UNIVERSIDADE FEDERAL DO MARANHÃO
Programa de Pós-Graduação em Ciência da Computação

MATHEUS CHAVES MENEZES

BRAIN-TO-BRAIN MAPPING: AN APPROACH TO SHARE NEURAL INFORMATION ON RATSLAM

São Luís
2019

UNIVERSIDADE FEDERAL DO MARANHÃO – UFMA
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MATHEUS CHAVES MENEZES

BRAIN-TO-BRAIN MAPPING: AN APPROACH TO SHARE NEURAL INFORMATION ON RATSLAM

SÃO LUÍS – MA
2019

MATHEUS CHAVES MENEZES

BRAIN-TO-BRAIN MAPPING: AN APPROACH TO SHARE NEURAL INFORMATION ON RATSLAM

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Alexandre César Muniz de Oliveira

SÃO LUÍS – MA

2019

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Menezes, Matheus Chaves.
Brain-to-brain mapping: an approach to share neural information on RatSLAM / Matheus Chaves Menezes. - 2019.
64 f.

Orientador(a): Alexandre César Muniz de Oliveira.
Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, Maranhão, 2019.

1. RatSLAM. 2. Robotics. 3. SLAM. I. Oliveira, Alexandre César Muniz de. II. Título.

MATHEUS CHAVES MENEZES

BRAIN-TO-BRAIN MAPPING: AN APPROACH TO SHARE NEURAL INFORMATION ON RATSLAM

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Dissertação aprovada em 22 de março de 2019.

Prof. Dr. Alexandre César Muniz de Oliveira
Orientador

Prof. Dr. Paulo Rogério de Almeida Ribeiro
Universidade Federal do Maranhão

Prof. Dr. Areolino de Almeida Neto
Universidade Federal do Maranhão

Prof. Dr. Edison Pignaton de Freitas
Universidade Federal do Rio Grande do Sul

SÃO LUÍS – MA
2019

Dedico este trabalho aos meus pais, Jorge e Coracy. Meu amor por vocês será eterno.

AGRADECIMENTOS

Agradeço primeiramente e principalmente aos meus pais. Sem eles, eu não seria quem sou, e muito menos chegaria aonde cheguei. Não cabe neste agradecimento o amor que sinto por eles, pois são as pessoas mais importantes da minha vida. Agradeço pelo esforço, paciência e especialmente pelo incentivo ao estudo, mostrando a mim que o conhecimento é a luz em meio à escuridão.

Agradeço também a minha família, pois sempre acreditaram em mim e me incentivaram a fazer o bem.

Agradeço aos meus professores Alexandre César, Paulo Rogério, Mauro Muños, Areolino de Almeida e Edison Pignaton. Além de possibilitarem a realização deste trabalho, ajudam na minha formação acadêmica e são exemplos de profissionais para mim. Agradeço em especial ao professor Alexandre por todas as oportunidades oferecidas dentro da Universidade. Obrigado por me acolher desde o primeiro período e ter colaborado tanto para meu crescimento acadêmico e pessoal.

Agradeço a todos os meus amigos, tanto aos meus velhos companheiros de ensino médio do IFMA, quanto aos meus companheiros da UFMA e do LACMOR. Obrigado pela ajuda e conversas jogadas fora durante esses longos anos. Um agradecimento especial aos meus amigos Marcos, Gilvan, Chrystian, Italo, Mellanie, Julianne, Dayson, Luann, Eduardo, Alex, Fernando, Felipe, Savanna, Dantas, Laine e Thayrê.

Agradeço em especial a Amanda Paixão pelo companheirismo ao longo desta nossa jornada. Obrigado por ouvir minhas loucuras que agora estão neste trabalho como resultados. Você será sempre importante na minha vida.

Agradeço à minha turma de Taekowndo pelos ensinamentos que vivi e vivo até hoje. A filosofia de vida nunca morrerá.

Agradeço à UFMA, à CAPES/BRASIL e à FAPEMA, bem como seus colaboradores. Os suportes estruturais e financeiro foram fundamentais para o incentivo acadêmico e realização deste trabalho.

Por último, agradeço à você que estiver lendo esta dissertação, pois quando partimos, o nosso legado é o que nos mantém vivos.

“É apenas isto: se você vai ser humano, tem um monte de coisas no pacote. Olhos, um coração, dias e vida. Mas são os momentos que iluminam tudo. O tempo que você não nota que está passando… é isso que faz o resto valer.”

Neil Gaiman

ABSTRACT

Several robotic applications are better performed by systems with multiple robots rather than only one, for example, to explore large areas in time-critical post-disaster search and rescue missions. These advantages can be due to the division of activities, cost and time reduction. Simultaneous Localization and Mapping (SLAM) plays a central role in exploring unknown environments. RatSLAM, which is based on the navigation system present in the hippocampus of rodents' brain, has been widely used on video-based SLAM applications. In RatSLAM, neural information is defined as experiences, which associates characteristics of the environment and movement in a unique representation on the map. This work presents an approach to share neural information on RatSLAM, named Brain-to-brain mapping, in which experience from partial maps are shared by various robots to cooperatively construct a map of the entire environment. The first step to share neural information is to connect different instances of RatSLAM through a merge mechanism, specific for RatSLAM. To perform the merge, it is necessary that the robots pass through at least a common place among them and acquire the same experience about the common place. The merge enables all robots to know about their experiences (*pose cells*, *local view* e *experience map*) in a shared structure. Thus, the exploration robots can reuse learned experiences over the environment to improve their map procedure e.g. a robot can correct part of the map of another robot, while using shared information to improve its own map performing loop closure. Three experiments of different environments were carry out to validate the new approach: a simulated environment, a research lab, and a *dataset* used to validate the original work of RatSLAM. The results has showed that the final map built by robots with shared experience is visually similar (but not identical) to one built by one robot performing the same mapping task individually, i.e. without sharing information.

Keywords: Robotics, SLAM, RatSLAM

RESUMO

Diversas aplicações robóticas são melhor executadas por sistemas com vários robôs em vez de apenas um, por exemplo, para explorar grandes áreas em missões críticas de busca e resgate em cenários de pós-desastre. Essas vantagens podem ser devidas à divisão de atividades, redução de custos e tempo. A Localização e Mapeamento Simultâneos (SLAM) desempenha um papel central na exploração de ambientes desconhecidos. O RatSLAM, que é baseado no sistema de navegação presente no hipocampo do cérebro dos roedores, tem sido amplamente utilizado em aplicações SLAM baseadas em vídeo. No RatSLAM, a informação neural é definida como experiências, que associam características do ambiente e movimento em uma representação única no mapa. Este trabalho apresenta uma abordagem para compartilhar informações neurais no RatSLAM, chamado *brain-to-brain mapping*, no qual a experiência de mapas parciais é compartilhada por vários robôs para construir cooperativamente um mapa de todo o ambiente. O primeiro passo para compartilhar informações neurais é conectar diferentes instâncias do RatSLAM através de um mecanismo de fusão, específico para o RatSLAM. Para realizar a fusão, é necessário que os robôs passem pelo menos um lugar comum entre eles e adquiram a mesma experiência sobre o lugar comum. A fusão permite que todos os robôs saibam sobre suas experiências (*pose cells*, *local view cells* e *experience map*) em uma estrutura compartilhada. Assim, os robôs de exploração podem reutilizar experiências aprendidas sobre o ambiente para melhorar o seu procedimento de mapeamento, como por exemplo: um robô pode corrigir parte do mapa de outro robô, enquanto usa informações compartilhadas para melhorar seu próprio mapa fechando *loops*. Três experimentos de diferentes ambientes foram realizados para validar a nova abordagem: um ambiente simulado, um laboratório de pesquisa e um *dataset* usado para validar o trabalho original do RatSLAM. Os resultados mostraram que o mapa final construído por robôs com experiência compartilhada é visualmente semelhante (mas não idêntico) a um construído por um robô realizando a mesma tarefa de mapeamento individualmente, ou seja, sem compartilhar informações.

Palavras-chave: Robótica, SLAM, RatSLAM.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

PCN	*Pose Cells Network*
LVC	*Local View Cells*
EM	*Experience Map*
SLAM	*Simultaneous Localization and Mapping*

CONTENTS

1 INTRODUCTION

Robots are used in hazardous activities in order to preserve human safety. Several robotic applications are better performed by systems with multiple robots rather than only one, for example, to explore large areas. These advantages can be due to the division of activities, cost and time reduction. This can be seen, for example, in time-critical applications, such as post-disaster search and rescue missions. Moreover, specific tasks may require several agents to be performed, e.g. several robots searching land mines and cooperatively disarm them (PALMIERI et al., 2015) (RANGO et al., 2015).

Autonomous navigation is very active research area in robotics that is searching for alternative methods to provide intelligent ways to robots navigate in unknown environment. In order to perform the navigation, the robot needs to map the environment while it simultaneously localizes itself, which defines one of the most fundamental problems of robots named the *simultaneous localization and mapping* (SLAM) problem (DURRANT-WHYTE; BAILEY, 2006). Moreover, for systems with multiple robots, SLAM is performed cooperatively among them, where robots share information about the environment with the aim to merge individual maps into a global map (ALMEIDA et al., 2019) (RANGO et al., 2018) (PALMIERI et al., 2018) (LEE et al., 2012).

Bio-inspired approaches have been used to provide new insights to solve general problems. Their application for multiple robots have been proposed on activities that can be hostile or dull, dangerous and dirty (*3D*) for human interventions, e.g. search and rescue (SILVA et al., 2010) (BAKHSHIPOUR; GHADI; NAMDARI, 2017) (CAI; CHEN; MIN, 2013), autonomous exploration of hazardous areas (SHARMA et al., 2015) (RANJBAR-SAHRAEI et al., 2015), area surveillance (CALVO et al., 2011).

Additionally, research conducted in neuroscience shows that the hippocampus and entorhinal cortex play a role in mammalian' space navigation (HAFTING et al., 2005) (MCNAUGHTON et al., 2006). Neurons activate when the rats are at specific spatial places in an environment (O'KEEFE, 1976), or cells that fire when they heads face west (TAUBE; MULLER; RANCK, 1990), for example, provide evidences that part of the brain is specialized spatial sensing and help them locate themselves in that environment.

The knowledge acquired from study of hippocampus and entorhinal cortex for the tasks of navigating in individuals has been an inspiration source to develop robotics navigation algorithms (ZENO; PATEL; SOBH, 2016). One of them is the RatSLAM that is based on the navigation system present in the hippocampus of the rodent brain and solves the SLAM problem for indoor or outdoor environments using low-resolution camera as main sensor input (MILFORD; WYETH; PRASSER, 2004) (MILFORD; WYETH; PRASSER, 2006) (MILFORD; WYETH, 2008) (MILFORD; WILES; WYETH, 2010). Moreover, the RatSLAM algorithm associates a neural information (i.e. unique scenes from an environment and activation in a neural network) to specific location in this environment.

This work is inspired by the efficiency of navigation systems based on how mammals move through the environment and the advantages of reducing the time and effort of cooperative exploration in large environments.

1.1 Goals

So far, RatSLAM is design to operates with only one robot. Thus, the goal of this work is to propose an approach to share neural information with multiple robots on RatSLAM. Using a shared neural information structure, cooperative SLAM can be performed using the RatSLAM algorithm.

To verify the feasibility of cooperative SLAM in the RatSLAM by sharing neural information, the second goal of this work is to share raw video data between robots. Sharing video between RatSLAM algorithms verifies the behavior of the algorithm in processing videos from multiple robots without changing the basic structure in how it processes the videos.

In order to achieve the main goals of this work, the follow specifics objectives must be achieved as well:

- To review of RatSLAM inner structure;
- Development a approach to share videos on RatSLAM;
- Definition and develop a merge mechanism between RatSLAM algorithms;
- Development and test the shared structure of neural information for RatSLAM algorithms.

1.2 Contributions

The contributions of this work are as follows:

- A methodology for sharing video on RatSLAM instances.
- A mathematical and algorithmic modeling of a merge mechanism for RatSLAM structures.

1.3 Division of Study

The other Chapters of this thesis are organized organized as follows.

Chapter 2 presents the theoretical basis necessary for the construction of this thesis. The concepts related to SLAM, neurologically navigation and RatSLAM are introduced. Chapter 3 describes and details all the present methodologies of this thesis, as well as the proposals of experiments to test them. In Chapter 4, the results achieved in this work are presented and discussed. Finally, Chapter 6 discusses the final considerations and suggestions for future work.

2 THEORETICAL FOUNDATION

This chapter details the topics needed to understand the techniques used in the elaboration of the proposed method. The following sections address concepts about SLAM, neuro-inspired SLAM and navigation, which includes the RatSLAM.

2.1 Simultaneous Localization and Mapping

Simultaneous Location and Mapping (SLAM) is associated to the problem of robot navigating in unknown environments: a robot, while navigating in environment, should acquire a map thereof while locating itself in this map. The SLAM problem is related to artificial intelligence in mobile robotics, whereby its solution tries to provide the means to make a truly autonomous robot (DURRANT-WHYTE; BAILEY, 2006).

2.1.1 Mathematical Definition

According to (THRUN; LEONARD, 2008), SLAM is formally better described in a probabilistic terminology. A robot moves in an unknown environment, starting at a certain location with unknown coordinates. As the robot moves, it can sense and map the environment. Thus, SLAM aims to build a map while simultaneously determining the relative position of the robot in the map.

The robot location on a flat ground is denoted by x_t, which represents the two-dimensions coordinates in the plane and a single orientation value at a time t. The set of coordinates is defined by (THRUN; LEONARD, 2008):

$$X_T = \{x_0, x_1, x_2, ..., x_T\} \tag{2.1}$$

where T is a terminal time, and the initial location x_0 is unknown.

The odometry provides information about robot motion between two places. This data is given by u_t and characterized the locomotion of robot between time $t-1$ and t. The sequence of odometry data is given by (THRUN; LEONARD, 2008):

$$U_T = \{u_0, u_1, u_2, ..., u_T\} \tag{2.2}$$

As u_t is obtained from robot' wheel encoders, such measurements are noisy and do not sufficient to recover the past X_T from initial x_0 (THRUN; LEONARD, 2008).

The environment map is denoted by m, assumed time invariant, i.e. it models a static environment. If it is assumed that the robot takes one measurement at each point at a time, the relation between measurements of features in m and location x_T can be given by (THRUN; LEONARD, 2008):

$$Z_T = \{z_0, z_1, z_2, ..., z_T\} \tag{2.3}$$

After definitions, SLAM must to recovery a map model and the sequence of robot locations X_T from odometry and measurements of environment. Two main forms of the SLAM problem are defined: *full SLAM problem* and *online SLAM problem*. The *full SLAM problem* involves estimating the posterior robot pose over the entire robot path together with the map (THRUN; LEONARD, 2008):

$$p(X_T, m|Z_T, U_T) \tag{2.4}$$

The *online SLAM problem* attempts to recovery the actual robot location instead of entire path. This online form is defined as (THRUN; LEONARD, 2008):

$$p(x_t, m|Z_T, U_T) \tag{2.5}$$

The algorithm that addresses *online SLAM problems* is usually incremental and can process one data item at a time (THRUN; LEONARD, 2008).

2.1.2 SLAM with Multiple Robots

SLAM with multiple-robots finds motivation in the fact of mapping tasks run faster and more accurate with multiple agents rather than only one (SAEEDI et al., 2016). The missions are distributed among the robots, which must to coverage a respective subarea. In addition, various reference point measurements can correct noisy sensor readings. Cooperative SLAM (C-SLAM) (MOURIKIS; ROUMELIOTIS, 2006) is a framework aimed at solving the problem of SLAM for multiple robots, where these robots cooperate to estimate the poses of a certain robot and build the map of the environment. Moreover, in C-SLAM, local maps are merged in a global one.

The probabilistic definition of SLAM in Eq. 2.4 can be extended to multiple robots, such described in (SAEEDI et al., 2016). Considering two robots, for instance, and the robots' identification as alphabetical characters a and b, multiple-robot SLAM aims to calculate the posterior over poses of the robots and the map.

$$p(X_T^a, X_T^b, m | Z_T^a, Z_T^b, U_T^a, U_T^b) \tag{2.6}$$

Saeedi at al. (SAEEDI et al., 2016) also explains that sharing data among robots is a fundamental issue in multiple-robot SLAM, since they can share raw sensor data (HOWARD, 2006) or processed data (BIRK; CARPIN, 2006). The raw sensor data means that sensed information of environment, i.e. odometry or video readings, are not processed. This no-processed data implies in more flexibility, but require more processing power, bandwidth and reliable communication links among robots. Besides that, in this case, the robots may share redundant data. On the other hand, processed data are result of sensor readings processed through smoothing, filtering or other methods, i.e maps, poses of robots, etc. Sharing processed data needs less bandwidth, however, the performance depends much more of the quality of the shared data. Thus, the choice of the method depends on factors such as the available resources or the proposed application.

2.2 Neurobiologically Spatial Navigation

The hippocampus and entorhinal cortex play a role in mammals' navigation with specialized navigation neurons found on rats and humans brains (O'KEEFE; DOSTROVSKY, 1971) (O'KEEFE, 1976) (HAFTING et al., 2005) (O'KEEFE et al., 1998) (EPSTEIN et al., 2017). This neurons, or cells, include place cells, head direction cells and grid cells. With these cells, the cognitive map hypothesis proposes that the brain build a structure to represent the spatial environment to support memory and future decisions on navigation (EPSTEIN et al., 2017).

2.2.1 Place Cells

Place cells, discovered in rodents brains in 1971 by O'Keefe and Dostrovsky (O'KEEFE; DOSTROVSKY, 1971) are able of increasing firing rate at a specific location in the rodent's roaming area. The firing location is invariant to the head direction cells or pose of rodent's body (ZENO; PATEL; SOBH, 2016). The activity of place cell is guided mainly by visual cues acquired by the rat from the environment. However, even in complete darkness or with sudden change in the environment, the activity patterns in place cells keep changing when the rat moves, updating and representing animal's position. This behavior shows that self-motion cues are also used to perform path integration, similar to dead reckoning in SLAM context (SÜNDERHAUF; PROTZEL, 2010a).

2.2.2 Head Direction Cells

Head direction cells, first found in rats by James B Rank Jr. in 1984 and examined by Jeffery Taube in 1990 (TAUBE; MULLER; RANCK, 1990), represent the global orientation of the animal's head (SÜNDERHAUF; PROTZEL, 2010b), and are invariant to the place or animal's body. Each cell has a preferred direction where it is fired at a maximum rate in relation to the rat's head direction in horizontal plane, although they seem to fall into a finite set of directions (i.e. N, NE, SW coordinates) (ZENO; PATEL; SOBH, 2016).

2.2.3 Grid Cells

Grid cells, discovered by Edvard and May-Britt Moser in 2005 (HAFTING et al., 2005), are located in the entorhnial cortex . In comparison with place cells, grid cells show different firing behavior. While a place cell fires when the animal is at a specific location, a grid cell exhibit multiple firing fields in space, showing a structure in grid and a regular and periodic pattern (ZENO; PATEL; SOBH, 2016). In addition, grid cells also are influenced by visual landmarks (HAFTING et al., 2005). The grid-shaped firing field are maintained without any visual cue (MOSER; KROPFF; MOSER, 2008).

2.2.4 Neurologically Navigation for Mobile Robots

Computational models of navigation cells have enabled the creation of robotic navigation systems that use neural mechanisms and concepts to perform goal-directed tasks. The advantages of using these biological systems is related to the fact that living mammals apparently do not show inherent problems rise of error in sensors, besides, dynamics environments are not a problem in perform navigation (HASSELMO, 2018).

The work developed by Zeno at. al. (ZENO; PATEL; SOBH, 2016) covers the main neurologically inspired robotic navigation systems since 2000. One of the reviewed works is RatSLAM (MILFORD; WYETH; PRASSER, 2004), which emulates Place and Head Direction cells, combining then in a an structure called Pose Cells Network (PCN). The RatSLAM cognitive map representation is an experience map, which uses the PCN to create and maintain experiences acquired from environment.

2.3 RatSLAM

RatSLAM is a mapping and localization system inspired on computational models of the neural process underlying navigation in the hippocampus of rodents and the entorhinal cortex. It was first introduced in 2004 by Milford, Wyeth and Prasser (MILFORD; WYETH; PRASSER, 2004) as a new approach to solve the SLAM problem. Over time, RatSLAM has been enhanced to work with general real-world examples of localization and mapping of mobile robots using vision system as its main input sensor. The Fig. 1 shows the RatSLAM architecture found in recent literature (MILFORD; WILES; WYETH, 2010) (BALL et al., 2013), where there are three main modules of the architecture: Pose Cells, Local View Cells and Experience Map. Additionally, there is a Robot Vision System and a Self Motion Cues modules. The modules have the following function:

i The Robot Vision System module aims to acquire the images and send them to the other modules;

ii The Self Motion Cues module is responsible for retrieving the translational and angular velocities information from the robot odometry. These information can also be calculated from the images acquired by Robot Vision System module.

iii The Pose Cells network module is a three-dimensional Continuous Attraction Network (CAN) of units connected by excitatory and inhibitory connections where each cell represents the robot's pose x, y and θ on the ground.

iv Local View Cell module creates the local view cells when a new scene is seen. These local view cells are activated and injects activity inside the pose cells via an excitatory link when the scene is seen again by the robot;

v The Experience Map module generates a structured graph with Cartesian proprieties that is a representation of the topological map of the environment.

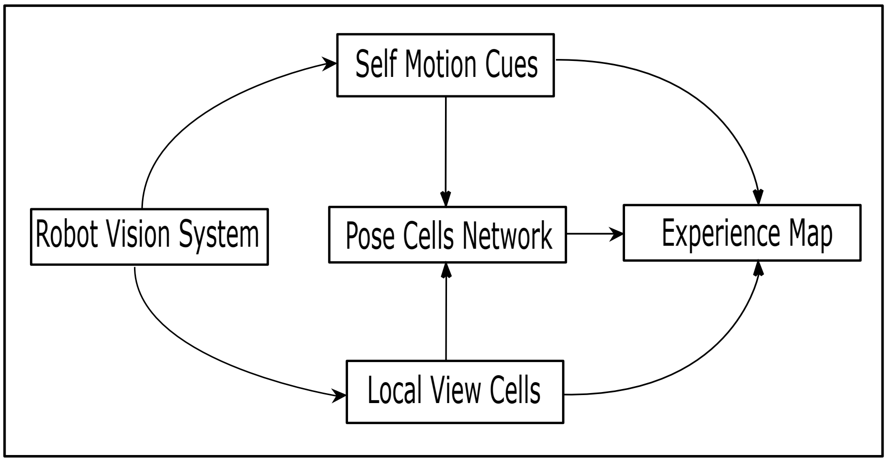


Figure 1 – RatSLAM architecture.

The first version of RatSLAM (MILFORD; WYETH; PRASSER, 2004) was different from the architecture shown in Fig 1. There was no implementation of an experience map, but a "goal memory" where path integration was performed. In (MILFORD; PRASSER; WYETH, 2005; MILFORD; WYETH; PRASSER, 2006), the *Experience Maps* (EM) are introduced as human-friendly representations of the environments. In addition, the continuity and local Cartesian properties of the experience maps allow them to be suitable for goal directed navigation. Large environments were mapped with RatSLAM in (PRASSER; MILFORD; WYETH, 2006) and (MILFORD; WYETH, 2008), where the first one used a 360 vision camera as main sensor, and the second mapped a neighborhood in Australia with only a notebook low-resolution camera as vision main input. In (MILFORD; WYETH, 2010), a robot works autonomously as delivery in a office for a two weeks period. This work adds a navigation system to RatSLAM. In (BALL et al., 2013), a RatSLAM implementation is developed to work within the Robot Operating System (ROS).

2.3.1 Pose Cells Network - PCN

PCN P is a continuous attractor network (CAN) configured in a three-dimensional prism as shown in Fig 2. CAN can be seen as a neural network of an array of cells equipped with weighted excitatory or inhibitory connections (MILFORD; WYETH, 2010). These connections cross the boundaries of the prism, allowing the network to function infinitely (with restrictions), but with a fixed dimensions sizes. Besides that, the CAN operates by varying the activity of the cells, rather than by changing the values of the weighted

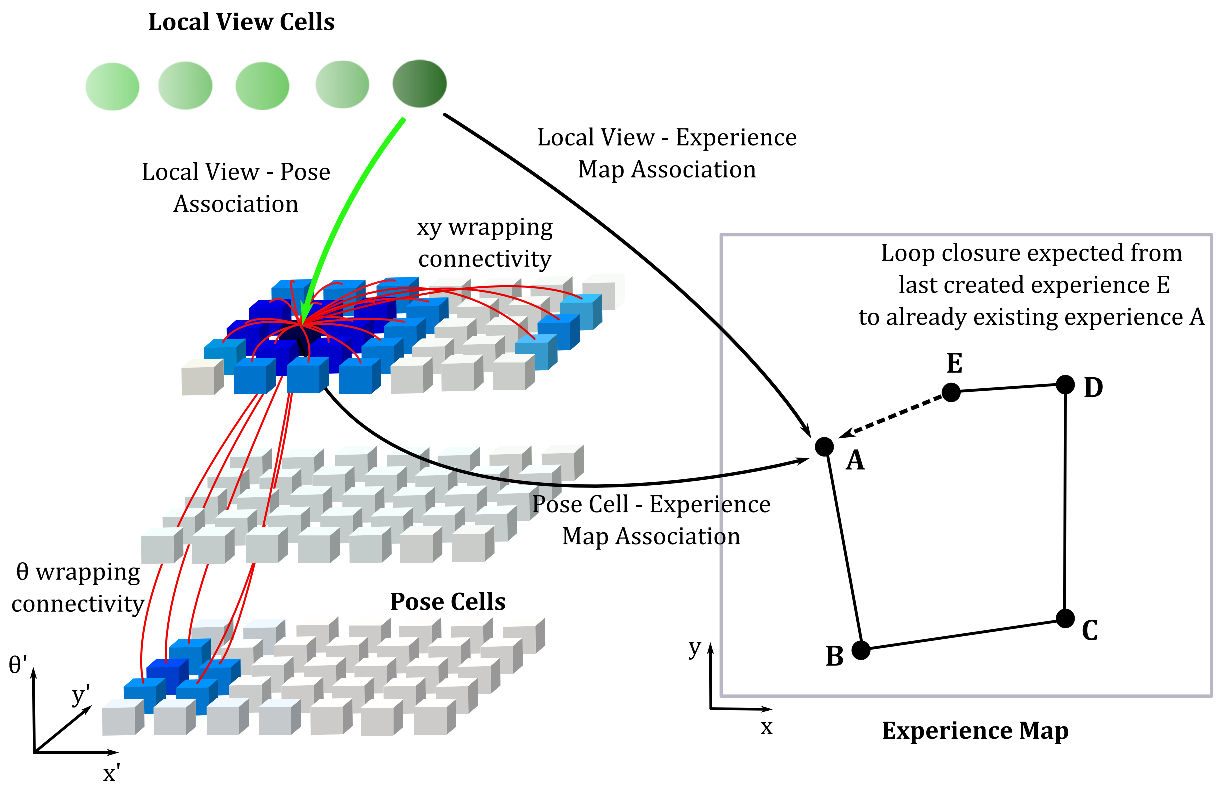


Figure 2 – Associations among Local View Cells, Pose Cells Network and Experience Map structures on RatSLAM.

connections (MILFORD; WILES; WYETH, 2010). The attractor dynamics of CANs, in its stable state, usually forms a single cluster of activated cells, known as an *energy packet* or *activity packet* (BALL et al., 2013). In addition, the cell array dimensions represent the three-dimensional information of x, y, and θ corresponding to the pose of a ground-based robot, and the centroid of the *activity packet* is the best estimate of the robot's current pose in the environment. This dynamical behavior is achieved with local excitation and globally inhibitory connectivity, as described by the distribution ε (MILFORD; WILES; WYETH, 2010):

$$\varepsilon_{a,b,c} = e^{-(a^2+b^2)/k_p^{exc}} e^{-c^2/k_d^{exc}} - e^{-(a^2+b^2)/k_p^{inh}} e^{-c^2/k_d^{inh}} \tag{2.7}$$

where k_p and k_d are the variance constants for place and direction respectively, and the a, b and c represents the distances between units in x', y', θ' coordinates respectively. Moreover, As show by the red lines in Fig 2, the connections of cells wrap across all six faces of the PCN, given the indices a, b, and c as (MILFORD; WILES; WYETH, 2010):

$$\begin{aligned} a &= (x' - i)(\text{mod } n_{x'}), \\ b &= (y' - j)(\text{mod } n_{y'}), \\ c &= (\theta' - k)(\text{mod } n_{\theta'}). \end{aligned} \tag{2.8}$$

The change of activity in a cell is given by (MILFORD; WILES; WYETH, 2010):

$$\Delta P_{x',y',\theta'} = \sum_{i=0}^{n_{x'}-1} \sum_{j=0}^{n_{y'}-1} \sum_{k=0}^{n_{\theta'}-1} P_{i,j,k} \varepsilon_{a,b,c} - \varphi \tag{2.9}$$

where $n_{x'}$, $n_{y'}$ and $n_{\theta'}$ are the network size in quantity of cells along each of the x', y' and θ' dimensions, and the φ amount creates the global inhibition. The information provided by odometry shifts activity in the PCN to represent robot's movement based on a nominal spatial size for each cell (BALL et al., 2013). The nominal size of a cell dictates the distance or the angle that it represents in real space. As an example, if a nominal cell size is $0.25m$ x $0.25m$, if the robot translate $0.25m$, the network activity will moves by unit in (x', y') plane.

2.3.2 Local View Cells - LVC

The LVC form an array of units, where each one represents a distinct visual scene in the environment. A LVC is created when a new visual scene is seen by the robot. A cell is then associated with the raw pixel data in that new scene (BALL et al., 2013). In addition, a short learning excitatory link β is built between the local view cell and the center of the dominant activity packet in PCN (BALL et al., 2013). This link is given by (MILFORD; WILES; WYETH, 2010):

$$\beta^{t+1}_{i,x',y',\theta'} = \max(\beta^{t}_{i,x',y',\theta'}, \lambda V_i P_{x',y',\theta'}) \tag{2.10}$$

where λ is the learning rate. When this scene is seen again by the robot, the local view cell is activated and it injects activity into PCN (x', y', θ') coordinates via its learnt excitatory link (BALL et al., 2013):

$$\Delta P_{x',y',\theta'} = \delta \sum_{i} \beta_{i,x',y',\theta'} V_i \tag{2.11}$$

where δ is the constant that determines the influence of visual landmarks on the robot' pose estimate (BALL et al., 2013). If a consecutive sequence of familiar scenes occurs in a correct order, the PCN will receive constant injection of activity in the pose which the scene was first viewed, resulting in the change of the dominant activity packet to this pose and re-localisation of the robot.

2.3.3 Experience Map - EM

The experience map is a two-dimensional graph map that combines pose cells and local view cells information to estimate the robot's pose. Each node in the experience map can be defined as a 3-tuple:

$$e_i = \left\{ P^i, V^i, \mathbf{p}^i \right\} \tag{2.12}$$

where P^i and V^i are the activity states in pose cells and local view cells respectively at the time the experience is created, and $\mathbf{p}^i$ is the robot's pose in experience map space. Moreover, a new experience is created when P^i and V^i are closely matched by the state associated with any existing experience. A score metric S is used to compare how closely the current pose and local view states match those associated with each experience, given by (BALL et al., 2013):

$$S^i = \mu_p |P^i - P| + \mu_v |V^i - V| \tag{2.13}$$

A link l_{ij} is created and saved when the robot moves from a previously active experience e_i to the new experience e_j (BALL et al., 2013):

$$l_{ij} = \left\{ \Delta\mathbf{p}^{ij}, \Delta t^{ij} \right\} \tag{2.14}$$

Where $\Delta\mathbf{p}^{ij}$ is the relative odometry pose between the two experiences, and Δt^{ij} is the time taken by robot to move between experiences. This temporal information is used to perform path planing from a specific experience to a desired goal using Dijkstra's algorithm and find the quickest path (BALL et al., 2013).

Until a loop closure process happens, the path generated by the robot is usually based in its odometry. The loop closure actives the robot re-localisation in the map and distributes the odometric error throughout the graph by a graph relaxation algorithm, changing the experiences' pose. Thus, the change in an experience's location is given by (BALL et al., 2013):

$$\Delta \mathbf{p}^i = \alpha \Big[\sum_{j=1}^{N_f} (\mathbf{p}^j - \mathbf{p}^i - \Delta \mathbf{p}^{ij}) + \sum_{k=1}^{N_t} (\mathbf{p}^k - \mathbf{p}^i - \Delta \mathbf{p}^{ki}) \Big] \tag{2.15}$$

in which α is a correction rate constant set to 0.5, N_f is the number of links from experiences e_i to other experiences, and N_t is the number of links from other experiences to experience e_i (BALL et al., 2013).

3 METHODOLOGY

In this Chapter, the sharing information approaches on RatSLAM are presented and detailed. In Sections 3.1 and 3.2, the materials and environments setups used to test approaches are presented, respectively. In Section 3.3.1, the video sharing approach is presented, as well as how it will be performed in experiment for validation in one of the already presented environments. In Section 3.3.2, the main approach of this work is presented, and similar to the previous Section, the experiments that the approach perform are also presented.

3.1 RoboDeck

RoboDeck is an open source robotic educational platform produced by the Brazilian company Xbot[1]. It was designed to promote educational development and research (XBOT, 2011). The educational plataform includes the mobile robot, an *Software Development Kit* based on the C/C++ language and a program for testing. The robot has four ultrasonic sensors distributed in the center of each side, and two infrared sensors in its below structure (Fig.3). These sensors are used to avoid collision. In addition, the robot has USB camera, accelerometer, compass, encoders in the motors of traction and GPS. Communication with RoboDeck is done via WiFi. In terms of computational processing, the robot uses a raspberry PI 3 model B, which has a quad-core ARM Cortex-A53 running at 1.2GHz and 1GB of RAM.

In this work, the robot camera is used as a vision system for the RatSLAM. Furthermore, the control of the robot's trajectory is done manually using its test code.

3.2 Environments

Three different environments are explored in this work: a) a video featuring a virtually generated environment (video #1); b) a video of a research Test Environment (video #2); and frames from the dataset environment called *"iRat"* (video #3), which is used to validate the OpenRatSLAM implementation (BALL et al., 2013). All three environments have different configurations, as well require different parameters for RatSLAM algorithm.

[1] http://www.xbot.com.br/

Figure 3 – Robodeck Plataform.

3.2.1 Virtual Environment

The virtual environment is an 360 degrees video that simulates a robot moving through designed environments, and it is a simple solution for testing different configurations of these environments, especially for vision-based SLAM algorithms.

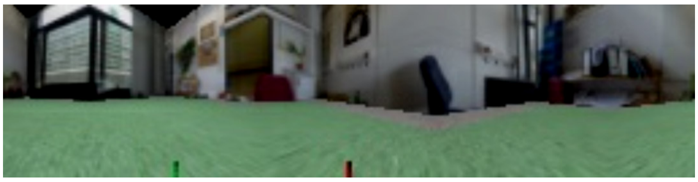
The Figures 4(a) and 4(b) show sample frames from the virtual environment and the path created in the video is an ellipse, as seen in Figure 4(c). In Video #1, the robot gives three full turns over the path, however the second and third turns are repetitions of the first one.

3.2.2 Test Environment

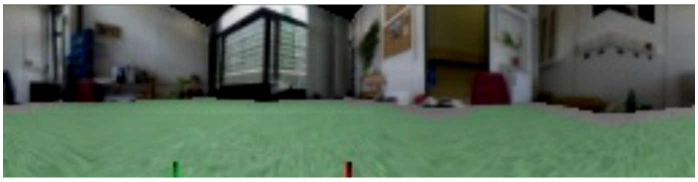
The test environment is a controlled area inside the laboratory. The laboratory is a rectangular indoor room (Figure 5), where two tables are placed in the center, serving as obstacles to be avoided, as well as to divide the lab in two subregions. In Figure 6(b), black regions are unreachable areas for the robot, due to the chairs scattered around the environment. Moreover, Video #1 was created during a RoboDeck tour along the lab.

3.2.3 *iRat* Environment

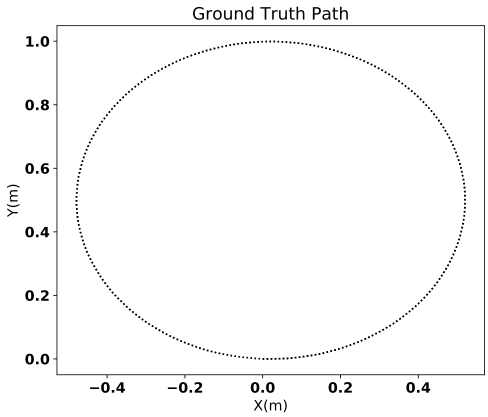
The *iRat 2011 Australia dataset* was obtained while a small mobile robot, similar in size and shape to a large rodent (iRat) (BALL et al., 2013), explored a road set based on a Australian geography (Figure 6). The iRat robot had been equipped with a overhead camera, able to provided images to extract ground truth information. The dataset is a



(a) Sample frames from video.



(b) Sample frames from video



(c) Ground truth of path traveled by virtual robot.

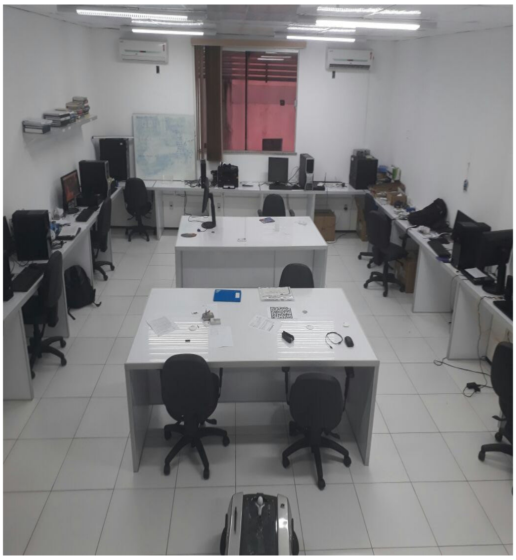
Figure 4 – Virtual Environment.

ROS bag file (QUIGLEY et al., 2009), including odometry and image information with approximately 16 minutes of duration. The iRat exploration was guided by a human, given directions to robot on which way to turn at each intersection (BALL et al., 2013).

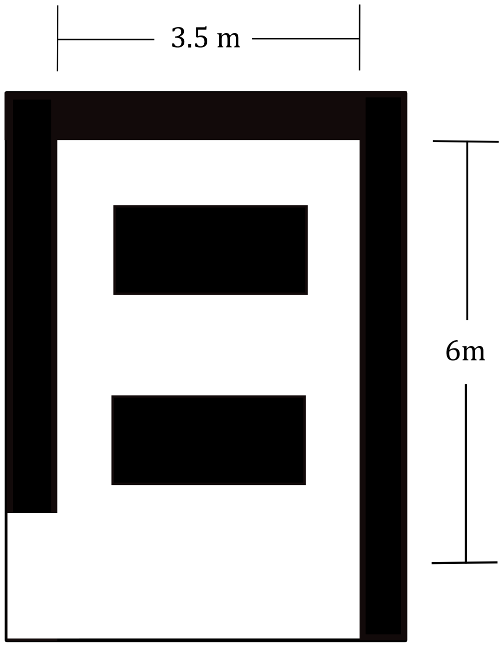
3.3 Sharing Information on RatSLAM

Sharing experience on RatSLAM can result in the *merge of minds* as long as computational procedures are able to manipulate data structures that represent knowledge about a given environment, producing a global shared structure of experiences from local experiences.

A first effort to share experiences is based on the mere distribution of video streams between robotic agents capable of individually processing and making available to others the environment data obtained from cameras. This first approach is called *Video Sharing for environment mapping*.



(a) Real room picture.



(b) Test Environment sketch, showing unreachable areas.

Figure 5 – Test Environment.

Then, a second approach is proposed to perform experience sharing by defining a new method for merging and sharing data among RatSLAM instances. This second approach is called *Experience Sharing*.

3.3.1 Video Sharing for Environment Mapping

Vision system provides a wealth of information, allowing greater versatility and providing important operations, such as detection and identification of objects. RatSLAM has been developed to deal with vision system as the main data entry.

In an exploration task, the time spent by the robot is related to the area explored by it, which means that the larger the area to be explored, the longer the time spent to complete the task. Thus, the advantage in sharing information among multiple exploration robots, initially, is to reduce the area of exploitation of each one and consequently, the time necessary to complete this task. For this, it is fundamental that each robot knows what has been exploited so that the task does not become redundant.

Sharing video between robots from previously exploited areas may prevent this environment from being exploited again. Figure 7 illustrates a scenario where a robot can continue its mapping task by recovering a video from other robot video without redoing the route. One possibility for this scenario would be the *Robot 1* choosing to

(a) A overhead view of *iRat* environment.

(a) Source: (BALL et al., 2013).

(b) Sample frames from dataset

(c) Sample frames from dataset

Figure 6 – *iRat* environment.

follow another path still unexplored. In RatSLAM, the visual reinforcement of locations is required for the algorithm to close loops and correct odometry errors that accumulate over time. Thus, sharing videos with the RatSLAM allows both the mapping time to decrease and the path correction to the robot. Therefore, sharing video does not impair the RatSLAM algorithm complexity.

This first proposed approach consists of sharing video stream between two or more robots under certain initial conditions. The robots are posed at same initial location and are driven to make different paths, returning to the same starting point. Each robot records a video stream along its specific path, producing a partial map of the environment. After that, the video streams can be shared among the robots to generate a complete map. The approach allows that the time spent to explore the whole environment can be reduced in the same proportion of robots involved in the exploration process.

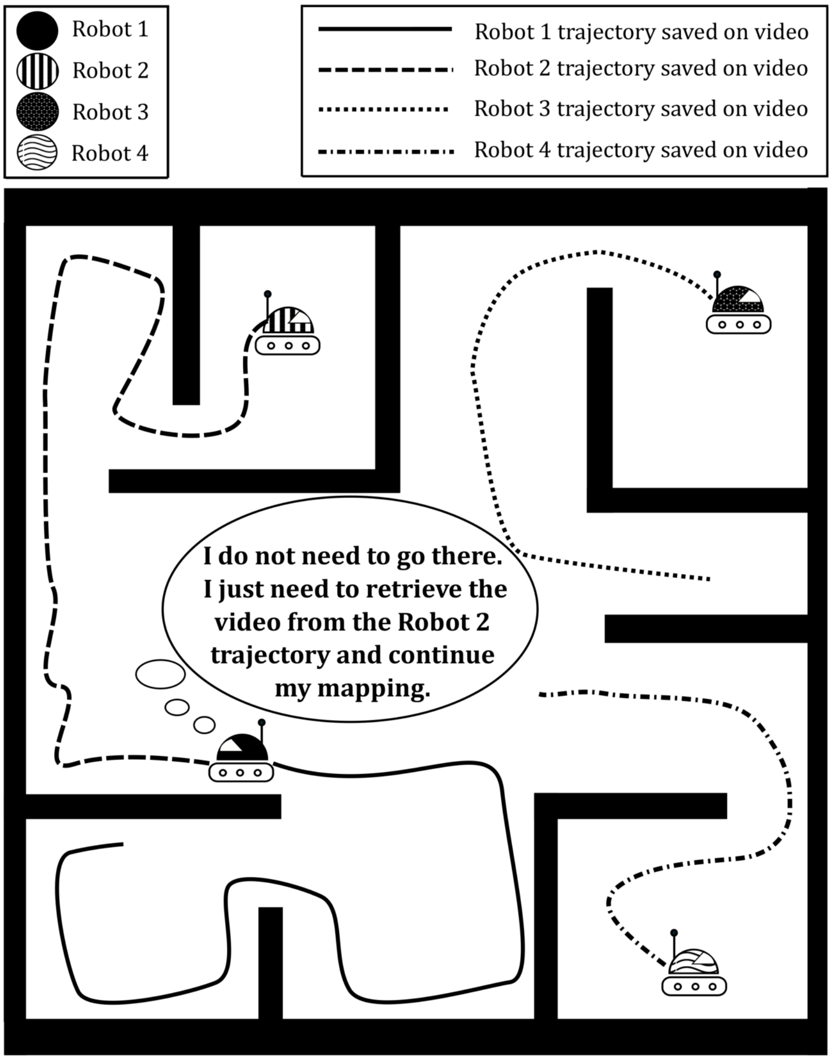


Figure 7 – A scenario for Sharing Video among robots.

Performing the Video Sharing Approach

Video Sharing consists of creating a complete map of a given environment from multiple video stream, shared by one or more robots. As can be seen in *Test Environment* depicted on Fig. 8, there are static obstacles that need the choice of two different courses to be performed by the robot. These choices are named “large” or “small” turn and they are represented by dashed lines and dot points, respectively. Since robots navigate following large or small turns, two different videos are produced suitable to be used for mapping and navigation processes.

Given two robots that depart from the same starting position producing two distinct video streams for later use in the global environment mapping, the recorded videos can be divided into parts according to the paths and directions performed by each

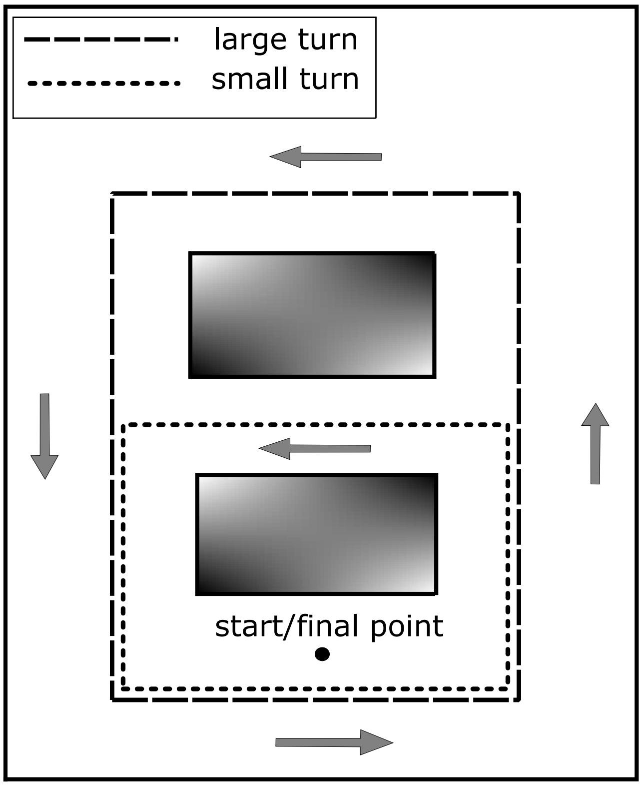


Figure 8 – Turn setup.

robots, so that snippets of videos are common to the two robots. Uncommon snippets can be shared and linked to common snippets to form a single video stream that can be used offline to generate a global map. This approach can be extended to an arbitrary number of robots, but in this work, only two robots have being considered.

Fig. 9 shows the methodology to map the full environment. First, a map is built using *video B* (dotted line), which is the small turn on Fig. 8, recorded by a first robot. Later, a second robot performs the large turn, which is *video C* on Fig. 9. Thus, this is the non-shared approach where each robot performs a specific turn. However, on the shared approach that is proposed in this work, a second robot only performs *video C'*, which is the non-common path between *video* B and C. This second robot uses *video* B^1 and B^2 from the first robot, which are extracted from *video B* and they represent the intersection between video B and C. The non-shared and shared approaches are summarized on Fig. 10a and Fig. 10b, respectively. Videos B^1, B^2 and C' were manually segmented and synced.

Additionally, a robot can play a previously recorded video and use RatSLAM algorithm to generate the map from it. Thus, a robot can perform what is called on this work a “virtual turn”, i.e. a video repetition of a real physical turn. This can be

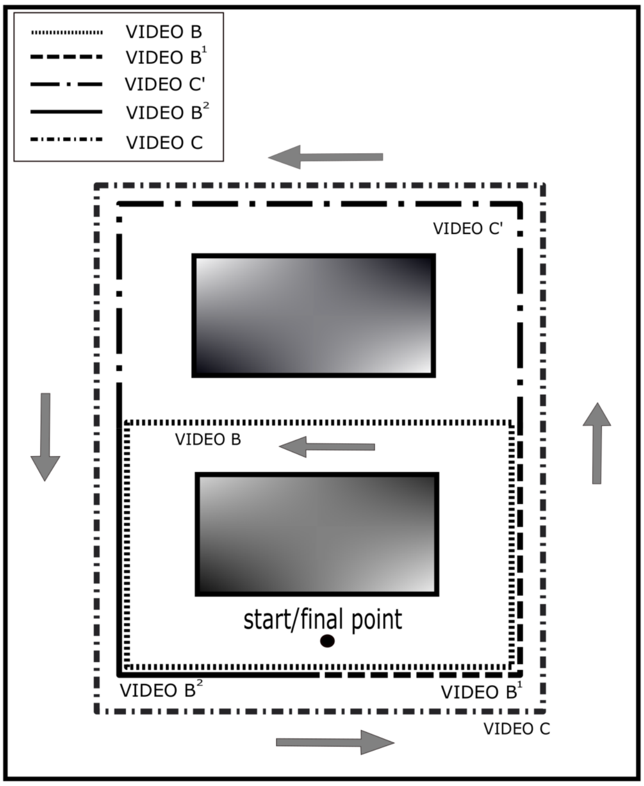
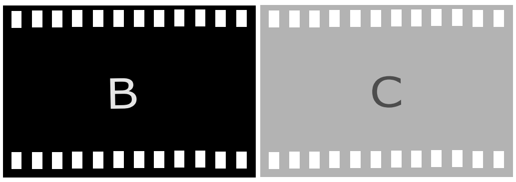
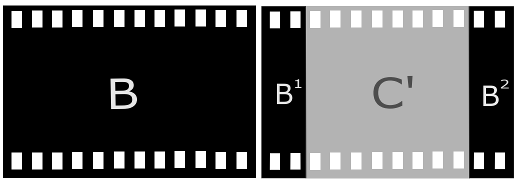


Figure 9 – Methodology to map the Test Environment.



(a) Non-shared approach



(b) Shared approach proposed on this work.
B, B^1, B^2, C and C' are delineated on Fig. 9.

Figure 10 – Non-shared and Shared approach summarized.

very useful for RatSLAM where redundant information, in this case the scene can be seen again, aiding to activate local view cells that inject activity inside the pose cell via excitatory link, therefore, a better map can be generated.

3.3.2 Experiences Sharing on RatSLAM

RatSLAM is an algorithm based on neural navigation mechanisms. In this context, neural information is equivalent to the experiences that the RatSLAM algorithm acquires in the SLAM process, associating and converting environmental and movement sensing to unique representations of locations in the environment. In this work, an experiment is a transformation of information from the environment that is processed in the RatSLAM and converted into information about the position of the robot and the path generated by it.

This approach proposes that in order to share experiences between robots, it is necessary to associate non-common experiences between them from a correspondence between common experiences, i.e the two robots must travel the same path and produce equivalent experiences over this path.

3.3.2.1 Problem Definition

Given two robots that are mapping the same environment, but by different paths, if a robot looks at a scene that was also seen by the other robot, there will be correspondence of experiences between them. This correspondence is the key to non-common experiences being associated from this common experience. The process that associates all experiments between the two robots is called *merge*. In the process of sharing experiences, the merge is used so that all the information acquired between the two robots is common among them, including the LVC and their associations for the PCN and experience map.

In the merging process, the LVC of the two robots are unified, as well as the associations between the PCN and the merge of their local maps. It is important to remember that each robot needs its own pose cell that represents the information where this robot is in space.

After this process, the two robots will be able to identify experiences already learnt by them and associate this information in the map of experiences, either creating new experiences or finding themselves in existing experiences.

Figure 11 exemplifies a situation where several robots are mapping different areas of the same environment. In Figure 11(a), it is shown that each robot has its own environment representation using RatSLAM, i.e. each has its own LVC array, its own

PCN and its own map. In figure 11(b), after the merge process, all LVC are placed in a single shared structure for all robots. In addition, each robot still has its PCN to represent its internal position in space, however, it is observed that the robots inject energy into their PCN through the LVC seen and learned by other robots. Finally, a global map of experiences is also generated from the merges of the local maps of the robots.

In the next section, the mathematical basis for the proposal is set out, where it is explained in more detail the steps of association between these experiences in common and how non-common experiences are co-related. To formulate this approach, only two robots are considered, but this approach serves for an arbitrary number of robots.

3.3.2.2 Mathematical Definition

The RatSLAM structure (BALL et al., 2013) is expressed by R by:

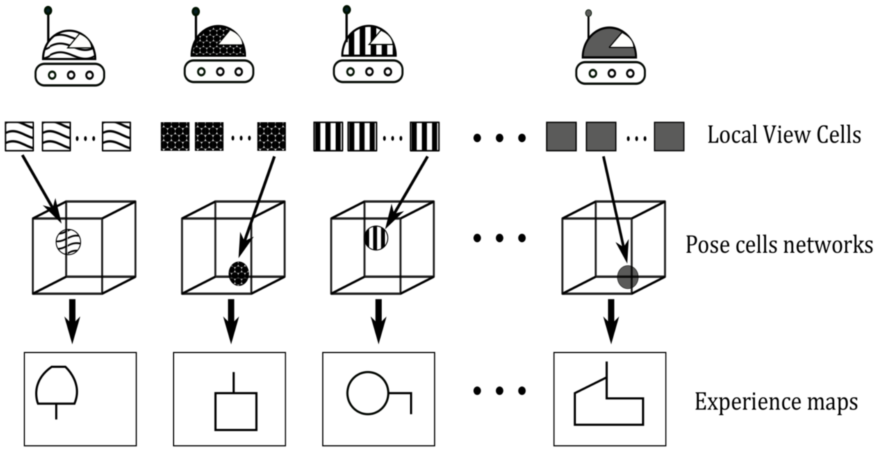
$$R = [P, X] \tag{3.1}$$

$$X = [V, B, G] \tag{3.2}$$

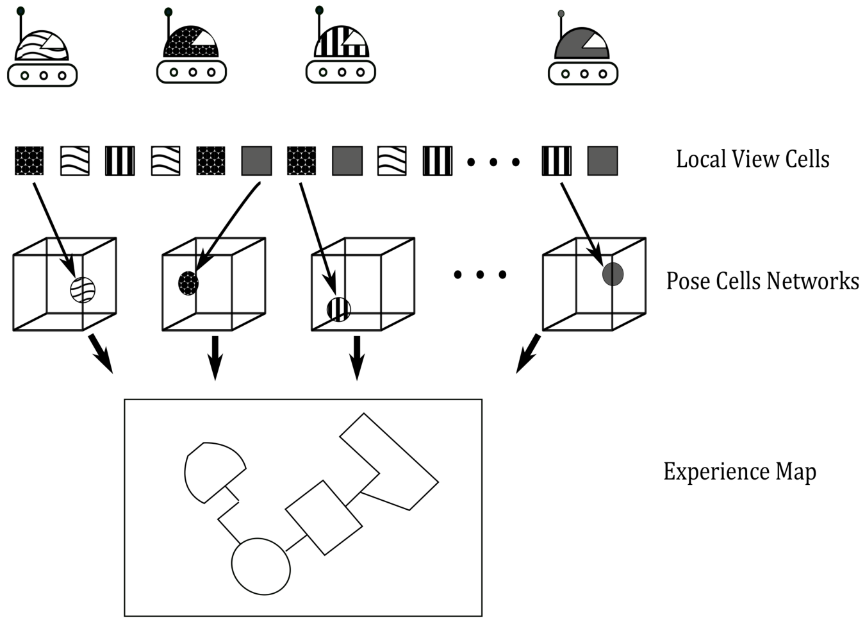
$$G = [E, L] \tag{3.3}$$

where:

- $P \in \mathbb{R}^{n_{x'} \times n_{y'} \times n_{\theta'}}$ represents the PCN of R. $n_{x'}, n_{y'}, n_{\theta'} \in \mathbb{N}$ are the previously defined CAN dimensions.
- X represents the structure that can be cooperatively built and shared.
- $V = \{V_i\}_{i=1,..n_v}$ is the set of Local View Cells, where n_v is the quantity of local view cells storage at LVC network
- $B \in \{0,1\}^{n_v \times n_x \times n_y \times n_\theta}$ stores the excitatory links β relations from V to P. If $B_{i,x',y',\theta'} = 1$ then V_i is connected to $P_{x',y',\theta'}$, otherwise they are not connected.
- G is the EM whose represent the map of environment, as well store experiences and links between experiences.
- $E = \{e_i\}_{n_e}$ is the set of experiences of G (Eq. 2.12). The n_e is the number of experiences in E;



(a) Each robot individually mapping a portion of the environment with the RatSLAM.



(b) The merge process applied to the robots, resulting in the union of all LVC and the merge of EM.

Figure 11 – Two SLAM scenarios with multiple robots.

- $L = \{l_{ij}\}_{n_l,n_l}$ is the set of links of G (Eq. 2.14). The n_l is the number of links conection;

For two RatSLAM structures used in robots A and B, it is used the index a to refer to robot A structures, i.e. P^a, X^a and V^a, and index b to refer the robot B. For shared structures, it is used *bar* element, i.e. $\bar{R}$, $\bar{P}$.

3.3.2.3 Merge of RatSLAM Inner Structures

The proposed RatSLAM merge procedure assumes that there is a robot A and a robot B moving in a unknown environment using the RatSLAM algorithm to map and situate they selves. There is a higher-level process comparing theirs LVC data capable to signal when some of them sees an scene already saw by the other. Robot A and robot B started in different unknown points from each other in the environment. Suddenly robot B perceives a scene and creates a *local view* just like some *local view* that robot A has already stored. Formally:

$$V_k^a = V_{n_{vb}}^b \tag{3.4}$$

Premises

Given the matched local views V_k^a and $V_{n_{vb}}^b$, the merge try to find the follow structure for R^a and R^b:

$$R^a = \{\bar{P^a}, \bar{X}\} \tag{3.5}$$

$$R^b = \{\bar{P^b}, \bar{X}\} \tag{3.6}$$

$$\bar{X} = \{\bar{V}, \bar{B}, \bar{G}\} \tag{3.7}$$

Where $\bar{X}$ is the merged structure between R^a and R^b. Note that it is assumed R^b had just found a local view $V_{n_{vb}}^b$ that matches some local view of R^a, named V_k^a. Thus R^a will be used as a base for the merge procedure. In other words, the operations of merge that modify the states of R (i.e. operations of translation and rotation apply in robot poses) will be made in R^b and will uses R^a as base. In addition, the comparison process that return a match between two local views is the same as used in RatSLAM algorithm, but to compare the local views of V^b with local views of V^a

Finding how to relate both RatSLAMs

Figure 12 illustrates how this approach merges two RatSLAM inner structures. The large arrow indicates the flow of time RatSLAM is going through. It is noted that each local view activates a region of the PCN via excitatory link. At any given time, a

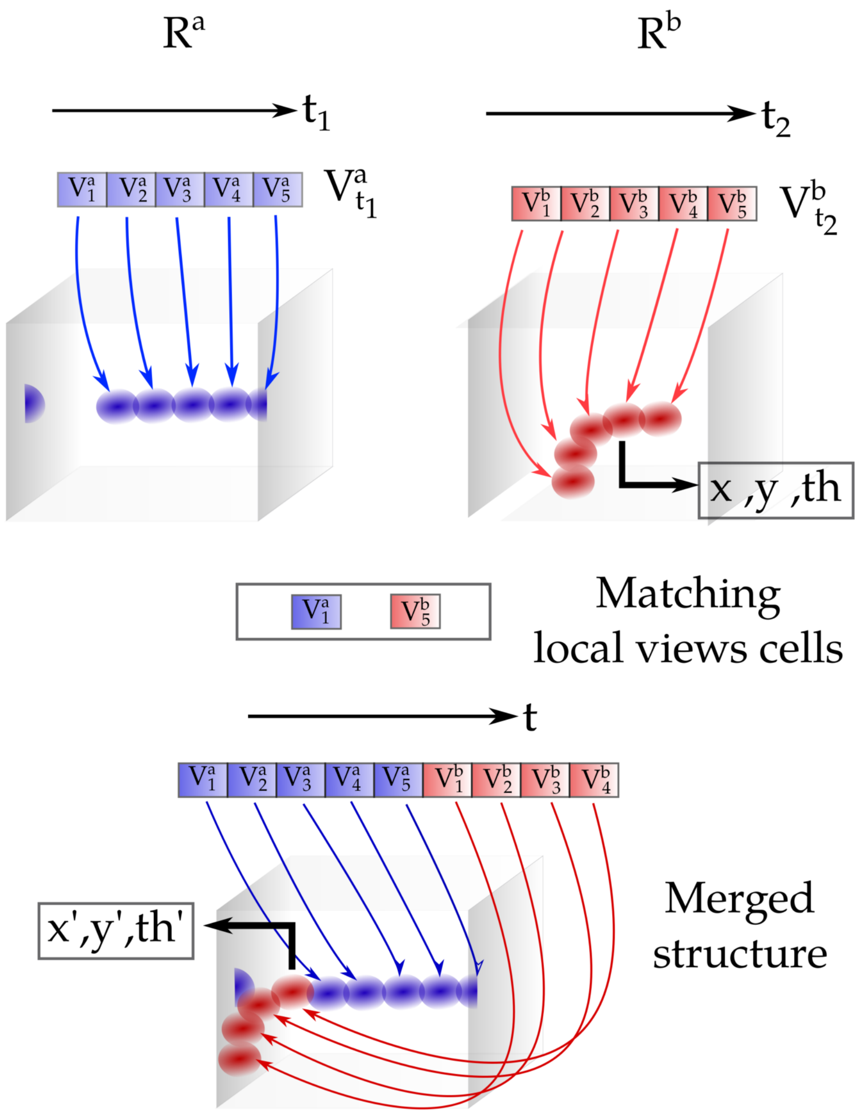


Figure 12 – Approach to merge two RatSLAM structures

correspondence between V_1^a and V_5^b is found. This correspondence considers that the PCN activation of these local views must also activate the same coordinates within the network. Thus, it is assumed that the pose cells activation links of R^b can be shifted to activate the R^a network. This shifting of coordinates is a function f that transforms the x, y and θ coordinates x', y' and θ':

$$\bar{\mathbf{c}} = f(\mathbf{c}) \tag{3.8}$$

where:

- $\bar{\mathbf{c}} \in \mathbf{N}^3$ are the resulting pose cells shifted coordinates;

- $\mathbf{c} \in \mathbf{N}^3$ are the original pose cells coordinates;
- $f : \mathbf{N}^3 \mapsto \mathbf{N}^3$ is the pose cell coordinates transformation.

Pose cells ($\bar{P}^a$, $\bar{P}^b$)

In RatSLAM algorithm, pose cells represent the probable pose the robot find itself at. When merging two RatSLAM it is important that each robot keeps its own assumption of its current pose.

Pose cells of R^a are kept as they were:

$$\bar{P}^a_{\mathbf{c}} = P^a_{\mathbf{c}} \quad , \forall \mathbf{c} \in n_c \tag{3.9}$$

Where $n_c = [x, y, \theta] \in \mathbf{N}^3$ is the set of coordinates of PCN P^a and P^b. Note that both PCN have the same dimensions, n_c.

Pose cells of R^b are displaced using the transformation (3.8). So:

$$\bar{P}^b_{f(\mathbf{c})} = P^b_{\mathbf{c}} \quad , \forall \mathbf{c} \in n_c \tag{3.10}$$

Resulting Local View Cells ($\bar{V}$)

The resulting local view, $\bar{V}$, embraces all local views of V^a and V^b, excepting the last seen view of V^b: $V^b_{n_{vb}}$, which is already represented by its counterpart in V^a: V^a_k. Thus:

$$\bar{V} = [V^a_1, .., V^a_{n_{va}}, V^b_1, .., V^b_{n_{vb}-1}] \tag{3.11}$$

Resulting Connections From Local Views to Pose Cells ($\bar{B}$)

The resulting connections, $\bar{B}$, should represent all relations established by B^a from V^a to P^a, and also the relations established by B^b from V^b to P^b. It must consider that V^a and V^b are merged to $\bar{V}$ and also that P^a and P^b had been transformed to $\bar{P}^a$ and $\bar{P}^b$, respectively.

$$\bar{B}_{i,\mathbf{c}} = B^a_{i,\mathbf{c}} \quad , i = 1, .., n_{va}, \ \forall \mathbf{c} \in n_c \tag{3.12}$$

$$\bar{B}_{j+n_{va},f(\mathbf{c})} = B^b_{\mathbf{c}} \quad , j = 1, .., n_{vb} - 1, \forall \mathbf{c} \in n_c \tag{3.13}$$

Note that $\bar{B} \in \{0, 1\}^{(n_{va}+n_{vb})\times n_c}$.

Resulting Experience Map ($\bar{G}$)

The experience map G is a graph responsible to store real world robot pose coordinates and its relations to RatSLAM pose cells and local views inner structures.

The construction of the resulting experience map $\bar{G}$ also takes R^a as reference. That is known (by: ref equation) that R^b is seen a scene and storing a local view ($V^b_{n_{vb}}$) that had already been seen by R^a, named V^a_k.

Experiences from R^a are simply transferred to $\bar{R}$ by:

$$\bar{e}_i = e^a_i \quad , \forall e_i \in E^a, \tag{3.14}$$

$$\bar{l}_{ij} = l^a_{ij} \quad , \forall l_{ij} \in L^a. \tag{3.15}$$

On the other hand, experiences in R^b must be transformed. From RatSLAM experience map construction algorithm (Eq. 2.12). The last created experience node of R^b is linked to its last seen local view $V^b_{n_{vb}}$. Thus, let:

$$e_q = [V_q, P_q, \mathbf{p}_q] = e^b_{n^b_e} \in E^b \tag{3.16}$$

represents the last created node experience of E^b where $V_{v_q} = V^b_{n_{vb}}$. By the merge condition (3.4), exists V^a_k such that $V^a_k = V_{v_q}$. The experience node of R^a associated with V^a_k is represented by:

$$e_r = [k, P_r, \mathbf{p}_r] \in E^a \tag{3.17}$$

The experience node e_q is associated with view $V^b_{n_{vn}}$ and the experience node e_r associated with view V^a_k. By condition (3.4), it is known that $V^a_k = V^b_{n_{vn}}$.

Poses $\mathbf{p}_q = [x_q, y_q, \theta_q]^T$ and $\mathbf{p}_r = [x_r, y_r, \theta_r]^T$ are used to find the geographical transformation $g : \mathbf{R}^3 \mapsto \mathbf{R}^3$ used to transform poses from E^b to E^a:

$$H = \begin{bmatrix} cos(\theta_q - \theta_r) & -sin(\theta_q - \theta_r) & 0 \\ sin(\theta_q - \theta_r) & cos(\theta_q - \theta_r) & 0 \\ 0 & 0 & 1 \end{bmatrix}; \tag{3.18}$$

$$g(\mathbf{p}) = H(\mathbf{p} - \mathbf{p}_q) + \mathbf{p}_r; \tag{3.19}$$

$$h(\mathbf{p}) = H\mathbf{p}. \tag{3.20}$$

Besides geographical informations, experiences of E^b also have references to pose cells and local views. These references are transformed using the f coordinates transformation function. Experiences and links from experience map E^b are added to the resulting experience map $\bar{E}$ by:

$$\bar{e}_{j+n_{ea}} = [V_{v_j+n_{va}}, f(P_j), g(\mathbf{p}_j)] \qquad , \forall e_j = [V_j, P_j, \mathbf{p}_j] \in E^b; \tag{3.21}$$

$$\bar{l}_{i+n_{ea},j+n_{ea}} = [h(\Delta \mathbf{p}^{(i+n_{ea})(j+n_{ea})}), \Delta t^{ij}] \qquad , \forall l_{ij} = [\Delta \mathbf{p}^{ij}, \Delta t^{ij}] \in L^b; \tag{3.22}$$

$$\bar{l}_{ij} = [0, 0] \qquad , i + j > n_{ea}. \tag{3.23}$$

3.3.2.4 Performing the Experience Sharing approach

To verify the neural information sharing approach on RatSLAM, three experiments setups are proposed, in which of them is related to the environments described in Section 3.2.

Virtual Environment Experiment

In this experiment, two robots A and B execute the mapping of the virtual environment (Section 3.2.1). Figure 13 illustrates the mapping proposal: a) robot A travels through part of the environment starting at the filled circle and ending at the white circle; b) after the robot A completes its mapping, robot B begins its process at filled pentagon, which is spatially close to where the robot A ended; c) robot B gives two complete turns on the environment and ends in the same location where it started.

Since robot B will pass through locations where it has already been mapped by robot A, it is expected that the merge of partial maps between robots A and B happens at where robot A started. Thus, robot B will use the experience map of robot A to navigate

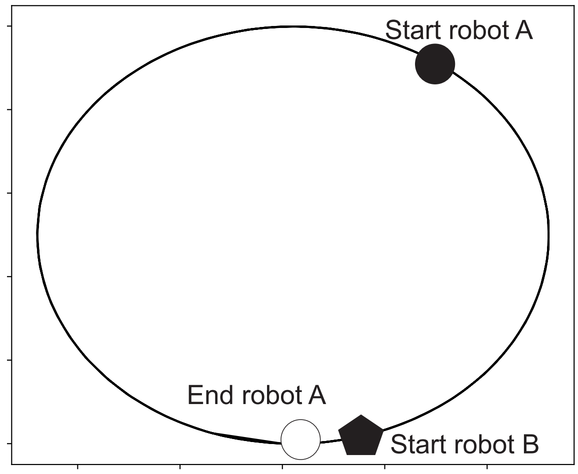


Figure 13 – Experiment to Virtual Environment.

this path already learned. Finally, robot B will close the loop at the transition from the end point of robot A (white circle) and at its starting point (pentagon filled).

Test Environment Experiment

The laboratory experiment is similar to that described in Section 3.3. Two robots A and B are used to map the environment shown in Figure 14. First, robot A performs its mapping by given two small turns (See Figure 8), starting and ending at the small circle. Then, robot B starts its SLAM at the white circle, performing the large turn and starting in a region that is not mapped by robot A. Robot B gives one complete large turn, then finish is mapping at the small end point.

The merge between robots A and B should happen in the intersection between two turns (left of Figure). Moreover, a loop closure should happen when robot B returns to its starting point.

iRat Environment Experiment

The *iRat* experiment is illustrated on Figure 15. Five robots are used to complete the total mapping of the environment. However, only one robot maps the environment at a time. Moreover, the map (and this RatSLAM data associated) produced by a robot is used as reference to the next one applies the experience shared approach. The filled circles that are in each line represent the starting point of the robot, as well as the filled

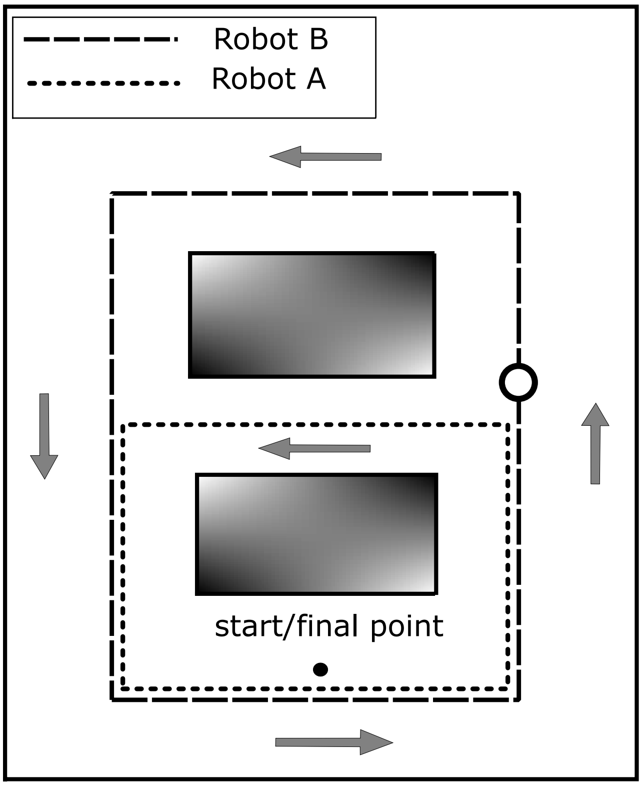


Figure 14 – Experiment to Test Environment.

triangle represents its end point of the mapping. Also, the arrows present near each line represents the direction in which the robot is moving. The procedure is as described in the following sequence:

- The robot A maps the external region of the circuit as shown in the solid line. It traverses a complete loop through the circuit to its starting point, where a loop closure must occur, plus the path to its end point.
- Robot B maps one of the internal turns of the circuit. It starts from a point not common with the map generated by robot A. At a certain point, the path that robot B runs is common already mapped by robot A. At this moment, it is expected that a merge between the experiences of the two robots where robot B will have access to robot A. After the merge between the two structures of the RatSLAM, robot B will continue its mapping until a complete turn and again closing a loop, plus the path to its stopping point. After robot B completes its mapping, the map generated will have both the experiences acquired by robot A and by robot B.
- The mapping procedure for robots C, D and E is the same as for robot B.

The contribution of each robot gradually builds the global map of the circuit. In the end, this global map is expected to be the merge of all individual robot maps.

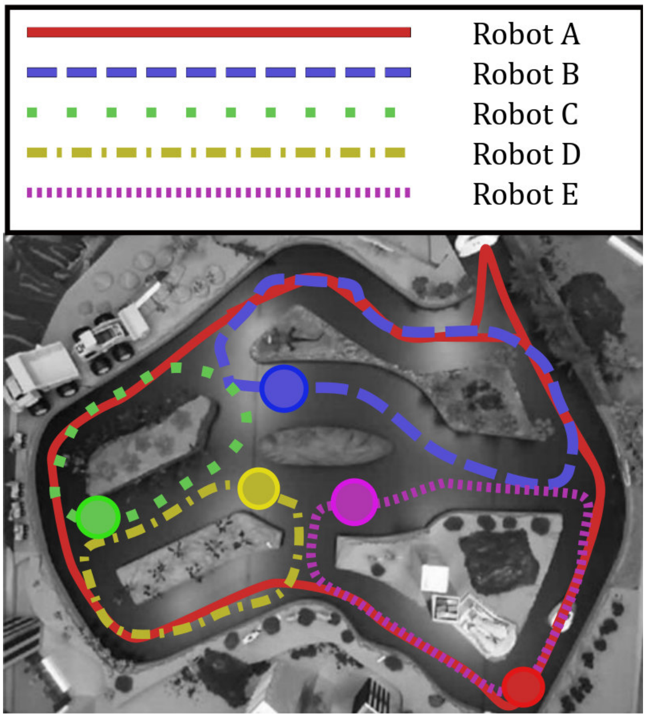


Figure 15 – Experiment to *iRat* Environment.

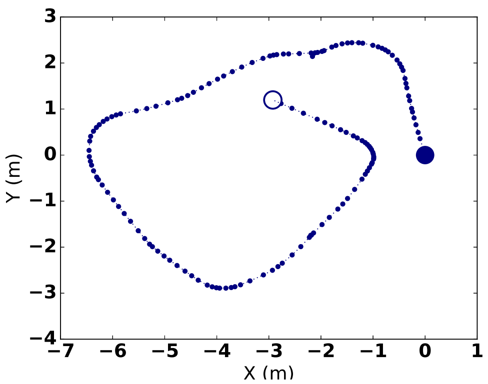
4 RESULTS E DISCUSSION

4.1 Video Sharing Approach

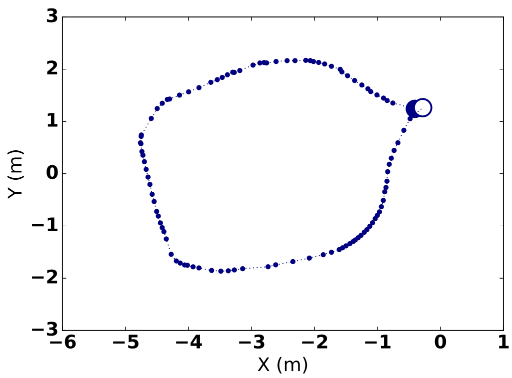
In order to verify whether sharing video between robots has advantages over the non-shared approach, a comparison between the maps generated with these two approaches is carried out. Furthermore, the time spent for each approach is taken into account. The results of this experiment were published in conference (MENEZES et al., 2018).

Laboratory Results

Figures 16(a) and 16(b) show the experience map from one "large" and "small" turns, respectively. The filled dot is the start point whereas the not filled dot is the end point. It is possible to note that on Fig. 16(a) there was no sufficient information to determine a loop closure in the map and this is differently of the result shown on Fig. 16(b).



(a) Large counterclockwise turn



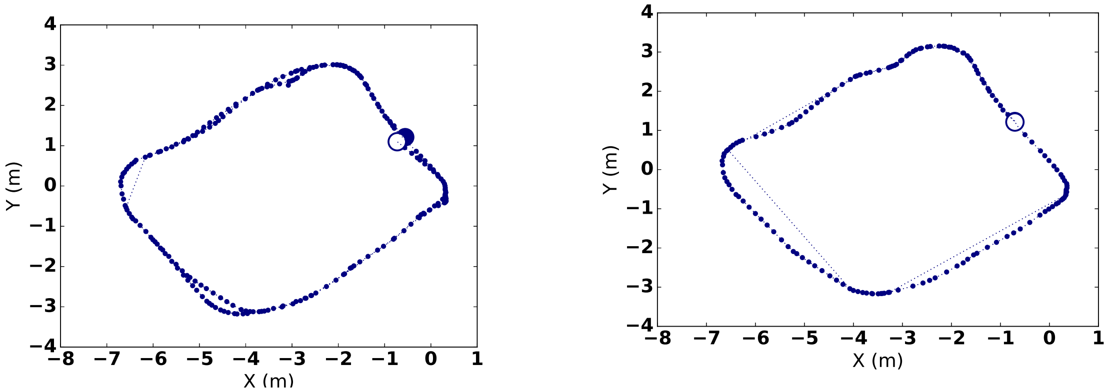
(b) Small counterclockwise turn

Figure 16 – Experience map of one counterclockwise turn.

The maps generated with the physical and virtual large turns are compared on Fig. 17(a) and 17(b), respectively. Similarly, a comparison between the maps generated with the physical and virtual small turns are shown on figures 18(a) and 18(b). As can been seen, for the large turn a better result was found with two turns, i.e. the map generated on figures 17(a) and 17(b) are better than the one on Fig. 16(a).

Even though Fig. 17(b) used only the recorded video employed to produce the results found on Fig. 16(a), it is noticed that the result on Fig 17(b) had a better experience map than the one on Fig 16(a). Additionally, there is no significant difference between Fig. 17(a) and 17(b) where the first means the physical turns whereas the second represents the virtual turns. It is worth mentioning that the virtual turns spent less time to collect the video data than the physical ones as shown on Table 1, that is, 83 seconds rather than 163 seconds, respectively.

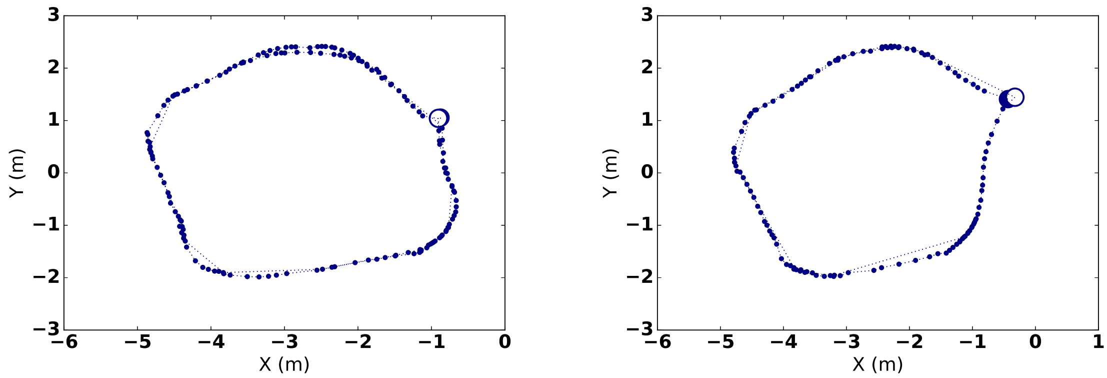
For the small turn it may not be necessary a second turn once the map displayed on Fig. 16(b) is reliable. However, in case it could be, the results on figures 18(a) and 18(b) show again that there is no difference between a physical and a virtual turn. Once more, as expected, this time is less for the virtual turn (60 seconds) than for the physical one (120 seconds) as demonstrated on Table 1.



(a) Two large physical turns.

(b) Two large virtual turns.

Figure 17 – Experience map of two large physical and virtual turns.

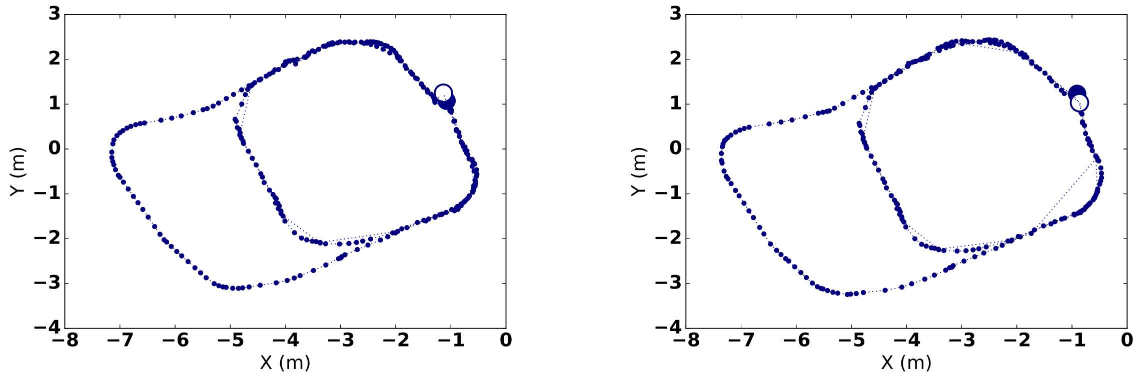


(a) Two small physical turns.

(b) Two small virtual turns.

Figure 18 – Experience map of two small physical and virtual turns.

Moreover, a comparison between the non-shared approach and the shared one for the full environment is shown on figures 19(a) and 19(b), respectively, which are the mapping using the approaches as depicted on figures 10(a) and 10(b). These results illustrate that there is no difference using the non-shared and shared approach. However, the shared approach spent less time, 174 seconds, than the non-shared one, 204 seconds, as presented in Table 1.



(a) Experience map using the non-shared approach

(b) Experience map using the shared approach

Figure 19 – Full mapping of Test Environment.

The recording time of each map is presented on Table 1. As the second virtual turn is a repetition of the first turn, two-turns virtual small/large have the same time of one turn.

Table 1 – Time for each turn and approach.

Recording video time (seconds)				
	Small	Large	Shared	No shared
One turn	60	83	-	-
Two physical turns	120	163	-	-
Two virtual turns	60	83	-	-
Complete path	-	-	174	204

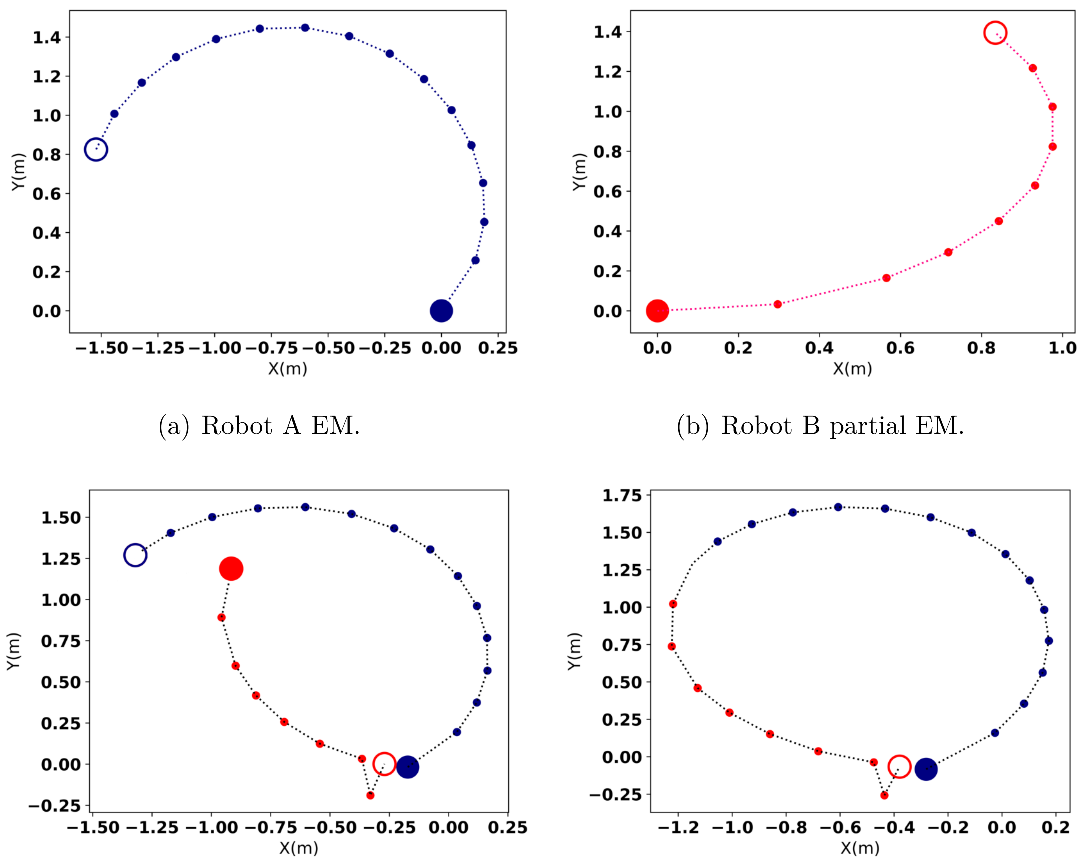
4.2 Experiences Sharing Approach

In order to verify the effectiveness of the experiences sharing approach, experiments of environment mapping were carried out using shared and non-shared approaches for comparison purposes.

4.2.1 Circle Environment Results

Figure 20 shows the shared information result of the virtual environment proposed in the Section 3.3.2.4 and performed by two robots A and B. Figure 20(a) presents the experience map (EM) obtained by the mapping of robot A, where the filled circle indicates its start point and the white circle the end of its trajectory. Also, note that there was no loop closure in robot A trajectory.

Figure 20(b) shows the partial EM created by robot B at specific time when it has found a correspondence between its and the local views of robot A. The merge process between the two RatSLAM mapping is illustrated in Figure 20(c). It is noticed that the end and start points of EM of robot A and B, respectively, are not connected, and this happens because robot B has not yet traveled over the environment enough to found a loop closure between the two EM. However, Figure 20(d) shows the final result of the process, by which robot B passed through the disconnected points of the maps and closed the loop.



(a) Robot A EM.

(b) Robot B partial EM.

(c) EM after the merge procedure.

(d) Final map of environment

Figure 20 – Virtual environment mapping.

Figure 21 shows the pose cells network (PCN) activations of the robots A and B. Figure 21(a) illustrates the center of the pose cells activations of robot A, while Fig. 21(b) shows the center of PCN activations of robot B before the process of merge of RatSLAM. Moreover, the activation on Fig. 21(b) next to the asterisk corresponds to the initial activation of the network before the robot begins the mapping. This activation is not considered in terms of implementation for the merge process. Figure 21(c) presents the PCN activations shared between A and B. It is observed that the robot B PCN activations had a shift to the bottom of the PCN activations of robot A in the shared PCN. These behavior is found according to Equation of Section 3.3.2.3, where the PCN of robot A remains the same after the merge procedure (Eq. 3.9), but the robot B activations change (Eq. 3.10). It is important to note that the Z axis on figures represents the rotation movement made by the robot in the real environment.

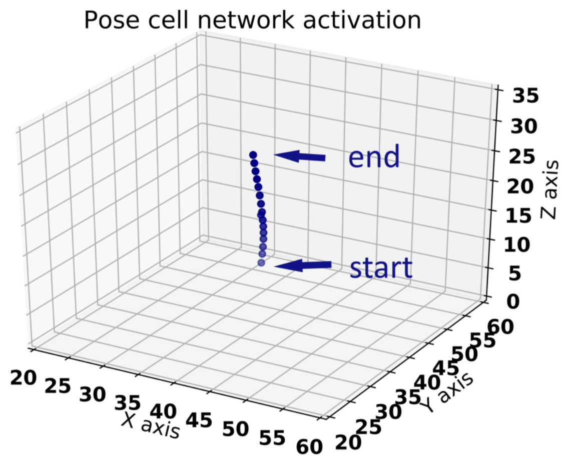
Finally, Figure 22 allows comparing results of the mapping with two robots and with a single robot. Figurs 22(a) and 22(c) show the EM and PCN using the proposed neural sharing respectively, as well as figures 22(b) and 22(d) show the EM and the PCN made up with only one robot. The EMs present a clear topological similarity. One can note that the PCN generated by the single robot has the same behavior of ascent in the axis of θ.

4.2.2 Test Environment Results

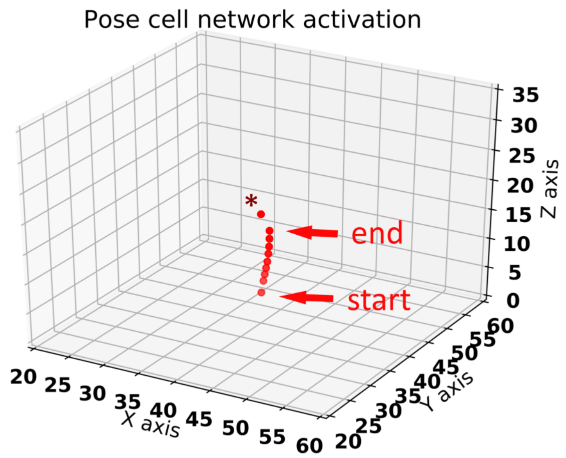
Figure 23 presents the results of the mapping in the research laboratory using two robots. Figure 23(a) shows the EM obtained by robot A after finishing its RatSLAM instance. Figure 23(b) shows the partial EM obtained by robot B while mapping the environment until the moment when a match was found between robot B local views in the LVC of the robot A.

The merge is then applied over experience maps of the robots (see Figure 23(c)). One can note that the map of robot B was translated and rotated to adapted to the map of robot A, following the Equations demonstrated in Section 3.3.2.3.

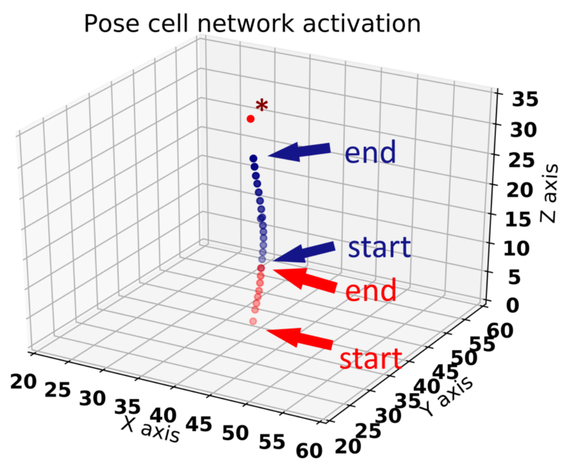
In Figure 23(d), while robot B continues its mapping, it is noticed that there is a correction in the EM generated by robot A. This correction shows that the merge between EM has influence on both robot's experiences. In this case, the experiences of robot B helped to correct the initial map of A. Finally, figures 23(e) and 23(f) illustrate



(a) Center of activation of robot A PCN.

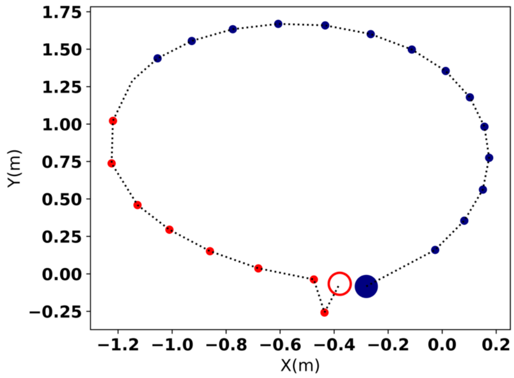


(b) Center of activation of robot A PCN.

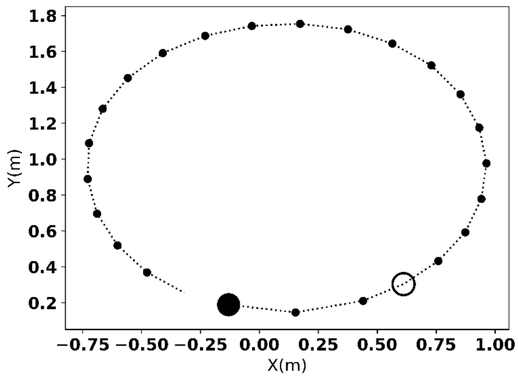


(c) Center of activation of shared PCN between robots.

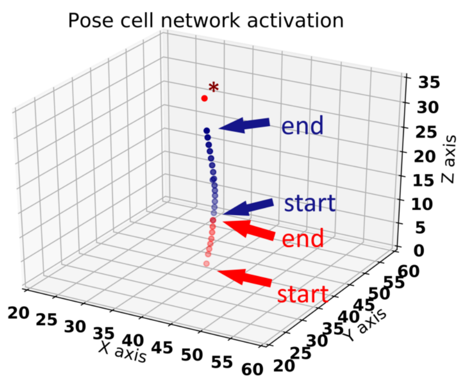
Figure 21 – Merge of pose cell activation.



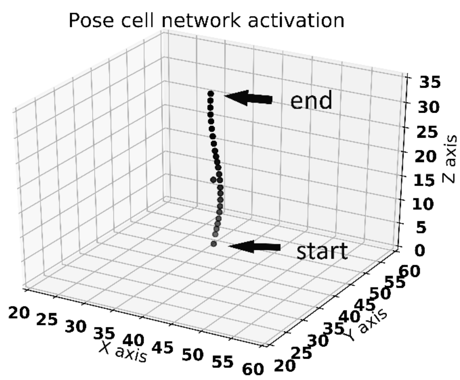
(a) EM from Shared Neural approach on RatSLAM.



(b) EM from only RatSLAM.



(c) PCN activations from Shared Neural approach on RatSLAM.



(d) PCN activations from only RatSLAM.

Figure 22 – Comparison between shared experiences approach and only one robot performing RatSLAM.

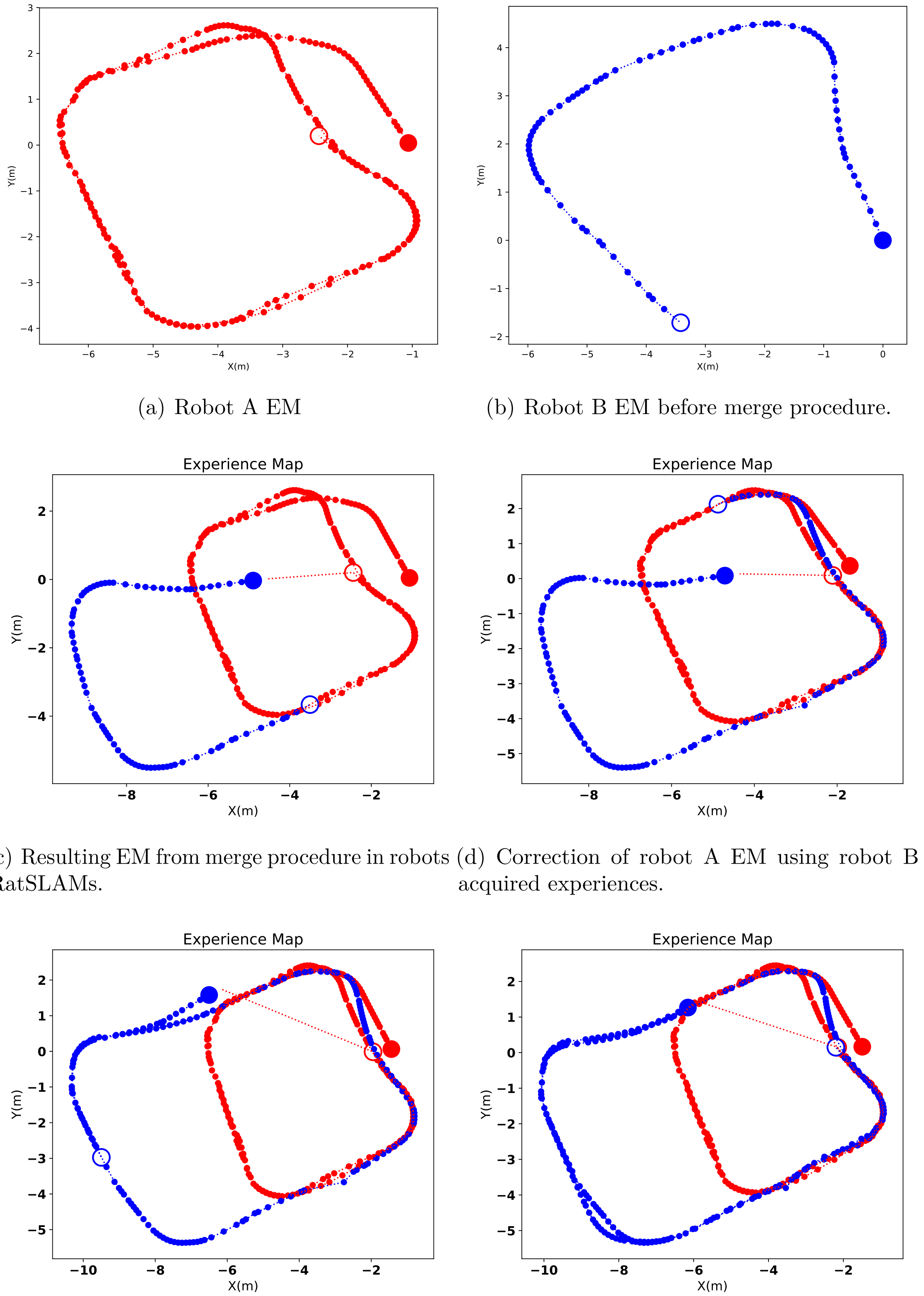
the loop closure between initials EMs of the robots A and B, and the final EM generated by the robots respectively.

4.2.3 *iRat* Environments Results

Figures 24, 25, 26 and 27 present the results of the *iRat* environment experiment. In all figures, the starting point of robot mapping is represented by the filled circle, while the end/stop point is showed as the white circle. Figure 24(a) shows the EM obtained by robot A. Moreover, Figure 24(b) shows the partial EM of robot B until the moment when its last experience has encountered a match with robot A experiences.

The result of the merge procedure between both robots is shown in Figure 24(c). Robot B then continued its mapping until it has closed a loop with its starting point, terminating its mapping in the blue white circle (see Figure 24(d)).

Figure 25(a) is the partial EM generated by both robots, which is considered



(a) Robot A EM

(b) Robot B EM before merge procedure.

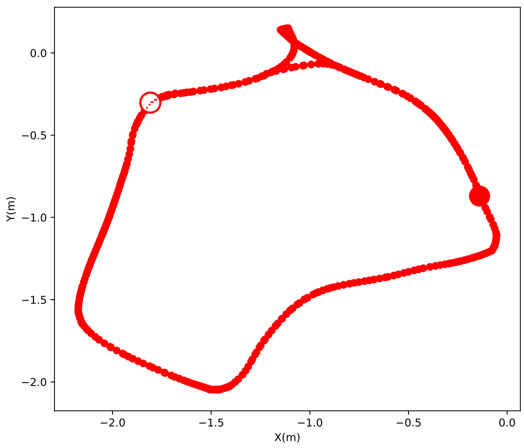
(c) Resulting EM from merge procedure in robots RatSLAMs.

(d) Correction of robot A EM using robot B acquired experiences.

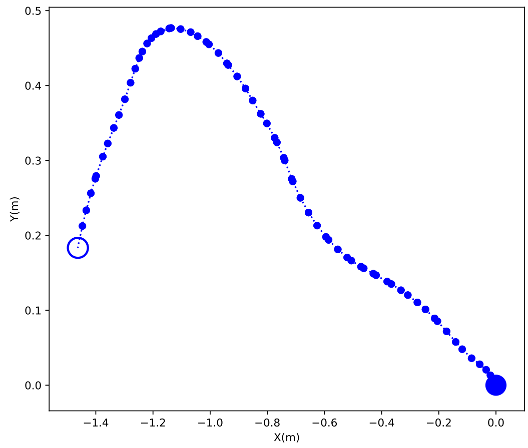
(e) loop closure between robots EMs

(f) Final EM generated by robots by shared information.

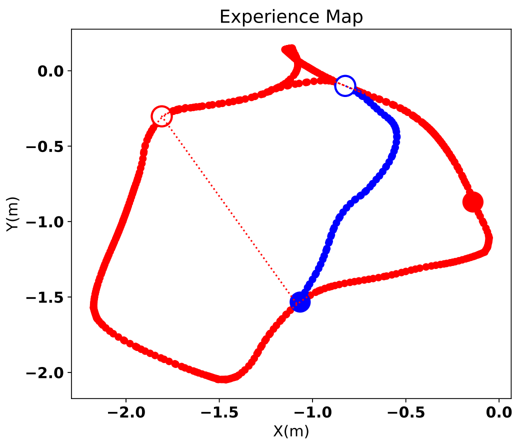
Figure 23 – Experience maps from shared experience approach of Test Environment.



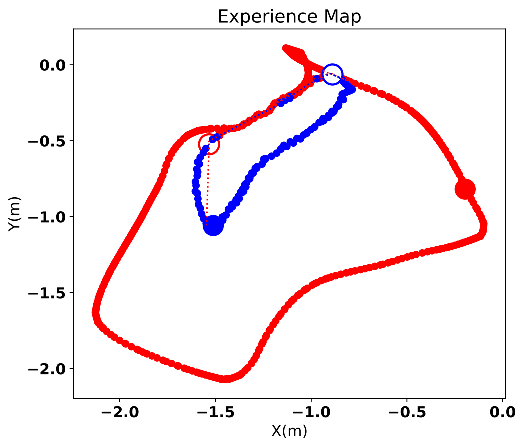
(a) EM from robot A RatSLAM.



(b) partial EM form robot B RatSLAM.



(c) EM resulting from merge procedure of robots A and B RatSLAMs.



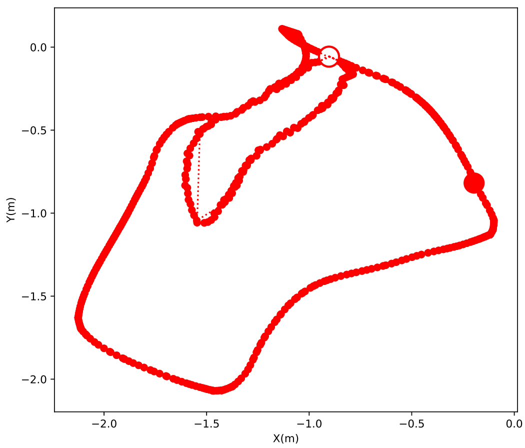
(d) EM cooperatively built using shared experiences from robots A and B.

Figure 24 – Experience map of *iRat* environment mapped by the robots A and B.

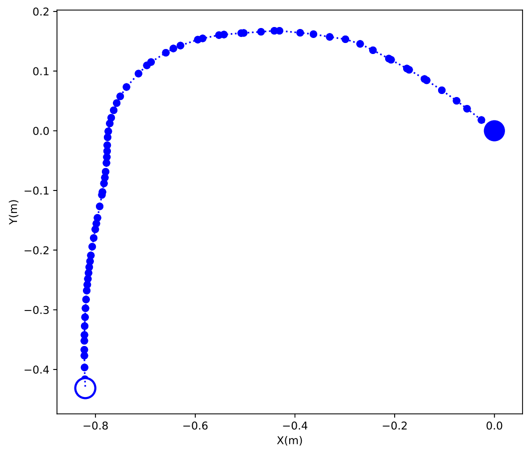
as the current map for sharing experiences with robot C. Similar to Figure 24, figures 25(b), 25(c) and 25(d) also show, respectively, the partial EM of robot C, the EM at the time after the merge procedure, and the final EM of the sharing experiences between the robots A, B and C.

Figures 26 and 27 follow the same sequence of figures 24 and 25. Figure 26 shows the EMs of robot D's mapping contribution and Figure 27 presents the EMs made up by robot E's mapping contribution. Thus, it is obtained the whole environment map.

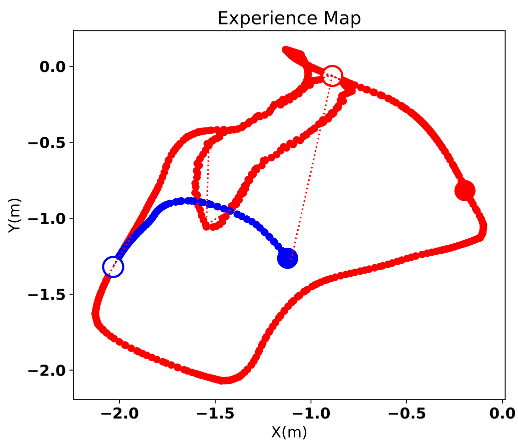
Finally, Figure 28 allows comparing a mapping of the same environment created with only one robot (Figure 28(a)) and the final map obtained with the shared neural information proposal among robots (Figure 28(b)).



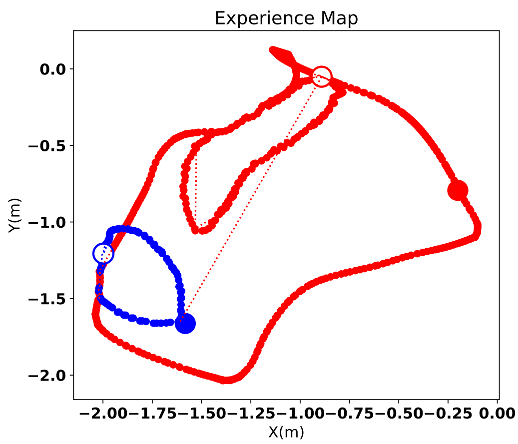
(a) EM from robot A and B RatSLAMs using shared experiences procedure.



(b) Partial EM form robot C RatSLAM.

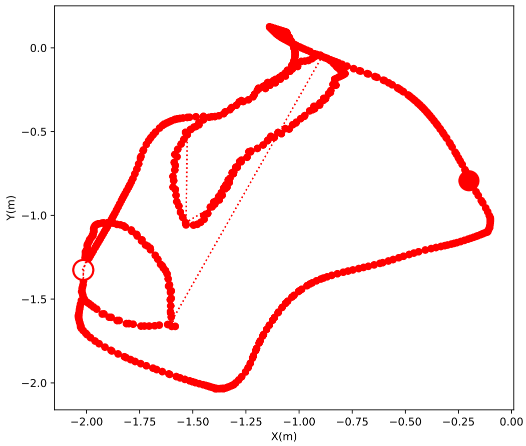


(c) EM resulting from merge procedure of robots A-B and C RatSLAMs.

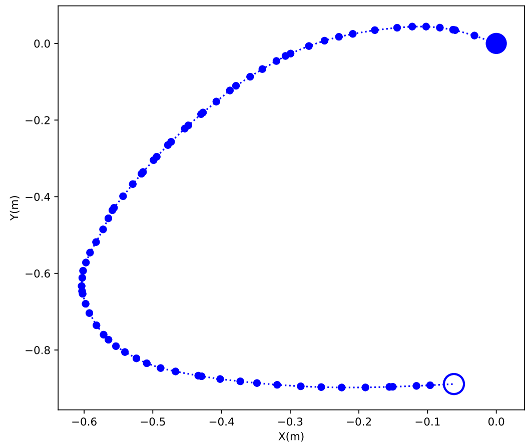


(d) EM cooperatively built using shared experiences from robots A, B and C.

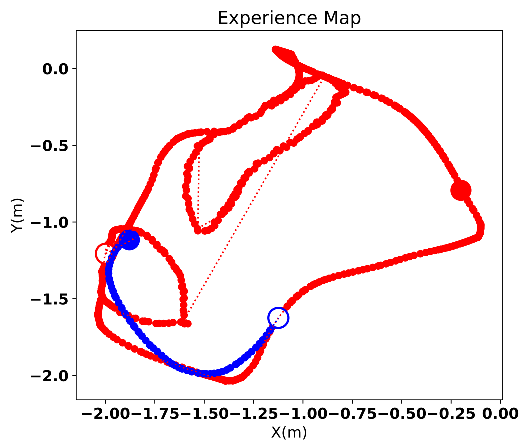
Figure 25 – Experience map of *iRat* environment mapped by the robots A, B and C.



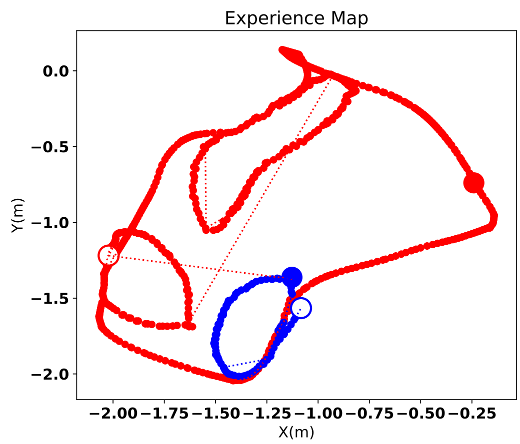
(a) EM from robot A, B and C RatSLAMs using shared experiences procedure.



(b) Partial EM form robot D RatSLAM.

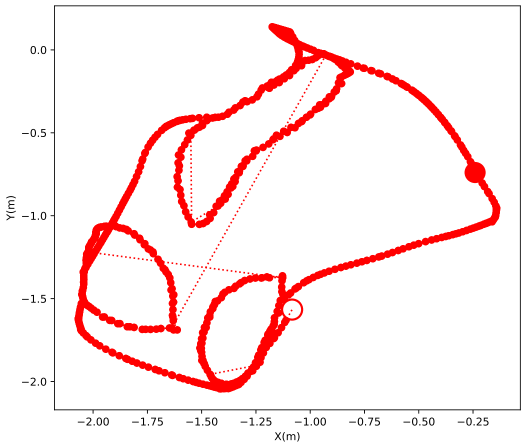


(c) EM resulting from merge procedure of robots A-B-C and D RatSLAMs.

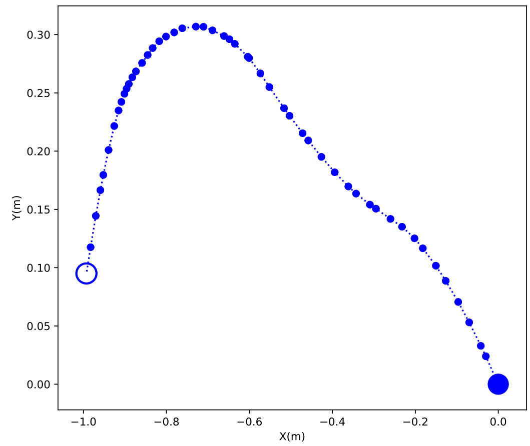


(d) EM cooperatively built using shared experiences from robots A, B, C and D.

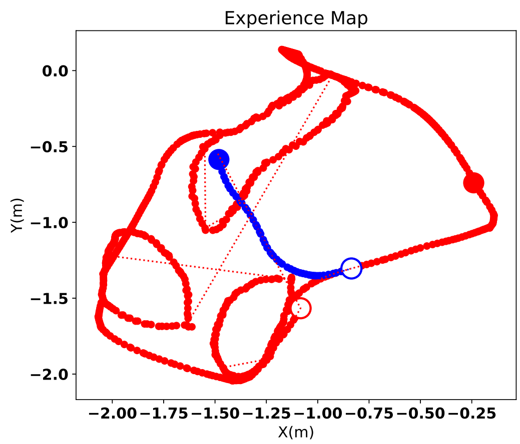
Figure 26 – Experience map of *iRat* environment mapped by the robots A, B, C and D.



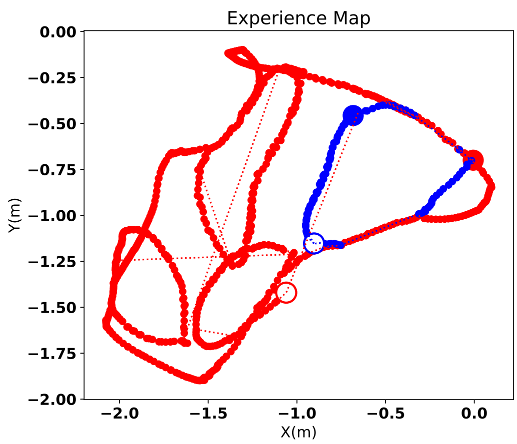
(a) EM from robot A, B, C and D RatSLAMs using shared experiences procedure.



(b) Partial EM form robot E RatSLAM.

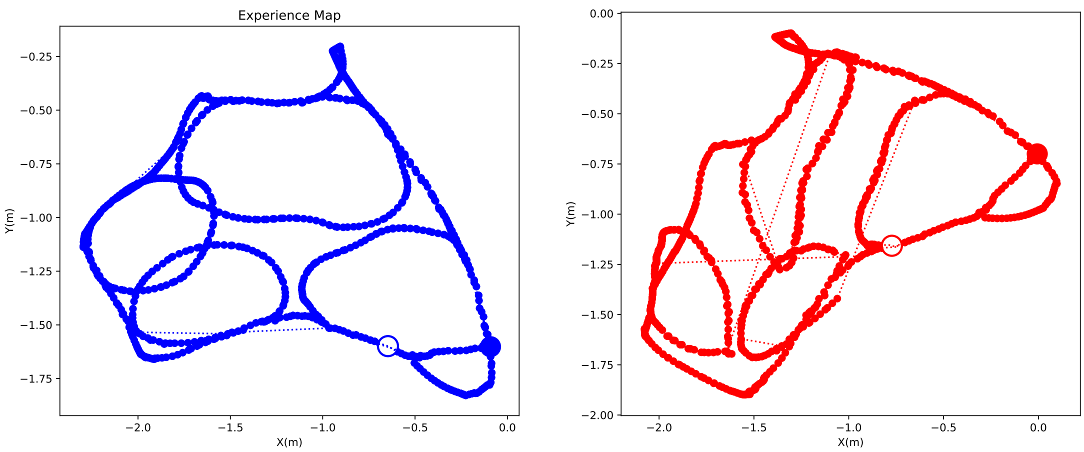


(c) EM resulting from merge procedure of robots A-B-C-D and E RatSLAMs.



(d) EM cooperatively built using shared experiences from robots A, B, C, D and E.

Figure 27 – Experience map of *iRat* environment mapped by the robots A, B, C, D and E.



(a) Experience map of one robot mapped with RatSLAM.

(b) Experience map of five robot cooperatively mapped with shared experience approach on RatSLAM.

Figure 28 – Comparison of Experiences maps generated by one robot and shared experience approach on RatSLAM.

5 CONCLUSION

This work proposed an approach to shared of neural information on RatSLAM. The proposal can be used to map large areas with multiple robots where each is responsible for mapping specific areas. The neural information in RatSLAM is given as an experience which associates local view cells captured by robot, activation in pose cells and geographic pose of robots path in experience map. The first step to share experiences was to develop an approach to share video information between robots that explore common paths. The merge mechanism that associates two similar experiences on RatSLAM was design and implemented. This mechanism was used to built the shared structure where all robots can have knowledge of its own and others experiences.

For shared videos approach, experiments were performed showing that it is possible to reuse videos with the same path to improve topological maps instead of making multiple physical tours. Furthermore, it also shown that maps of environments with common paths in common can be shared to create a global map without loss of topological map quality. Moreover, as an advantage, the time spent by the robot to map the full environment was shorter when information was shared than without sharing it.

For shared neural information approach, experiment of tree environments were performed showing that it is possible to multiple robots cooperatively map an environment using RatSLAM. The structure allowed robots to recover experiences saved by others. Thus, these robots could combine their maps with those of other robots, as well as use the experiments to correct their location within the environment. In addition, the topological maps generated resemble those made by a single robot, but only with statistical and experimental tests can you confirm if these maps are viable for navigation.

As future works, it is suggested that real-time mapping experiments be done with several robots at the same time in order to verify the feasibility of the proposal in real-world tasks, e.g search and rescue. Moreover, it is suggested that comparative navigation experiments with the proposal and one robot be done.

BIBLIOGRAPHY

ALMEIDA, J. P. L. S. de; NAKASHIMA, R. T.; NEVES-JR, F.; ARRUDA, L. V. R. de. Bio-inspired on-line path planner for cooperative exploration of unknown environment by a multi-robot system. **Robotics and Autonomous Systems**, Elsevier, v. 112, p. 32–48, 2019.

BAKHSHIPOUR, M.; GHADI, M. J.; NAMDARI, F. Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach. **Applied Soft Computing**, Elsevier, 2017.

BALL, D.; HEATH, S.; WILES, J.; WYETH, G.; CORKE, P.; MILFORD, M. Openratslam: an open source brain-based slam system. **Autonomous Robots**, v. 34, n. 3, p. 149–176, Apr 2013. ISSN 1573-7527. Disponível em: <https://doi.org/10.1007/s10514-012-9317-9>.

BIRK, A.; CARPIN, S. Merging occupancy grid maps from multiple robots. **Proceedings of the IEEE**, IEEE, v. 94, n. 7, p. 1384–1397, 2006.

CAI, Y.; CHEN, Z.; MIN, H. Improving particle swarm optimization algorithm for distributed sensing and search. In: IEEE. **P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on**. [S.l.], 2013. p. 373–379.

CALVO, R.; OLIVEIRA, J. R. de; FIGUEIREDO, M.; ROMERO, R. A. F. A distributed, bio-inspired coordination strategy for multiple agent systems applied to surveillance tasks in unknown environments. In: IEEE. **Neural Networks (IJCNN), The 2011 International Joint Conference on**. [S.l.], 2011. p. 3248–3255.

DURRANT-WHYTE, H.; BAILEY, T. Simultaneous localization and mapping: part i. **IEEE robotics & automation magazine**, IEEE, v. 13, n. 2, p. 99–110, 2006.

EPSTEIN, R. A.; PATAI, E. Z.; JULIAN, J. B.; SPIERS, H. J. The cognitive map in humans: spatial navigation and beyond. **Nature neuroscience**, Nature Publishing Group, v. 20, n. 11, p. 1504, 2017.

HAFTING, T.; FYHN, M.; MOLDEN, S.; MOSER, M.-B.; MOSER, E. I. Microstructure of a spatial map in the entorhinal cortex. **Nature**, Nature Publishing Group, v. 436, n. 7052, p. 801, 2005.

HASSELMO, M. E. Spatial and episodic memory. **Living Machines: A Handbook of Research in Biomimetics and Biohybrid Systems**, Oxford University Press, p. 271, 2018.

HOWARD, A. Multi-robot simultaneous localization and mapping using particle filters. **The International Journal of Robotics Research**, Sage Publications Sage CA: Thousand Oaks, CA, v. 25, n. 12, p. 1243–1256, 2006.

LEE, H.-C.; LEE, S.-H.; LEE, T.-S.; KIM, D.-J.; LEE, B.-H. A survey of map merging techniques for cooperative-slam. In: IEEE. **2012 9th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)**. [S.l.], 2012. p. 285–287.

MCNAUGHTON, B. L.; BATTAGLIA, F. P.; JENSEN, O.; MOSER, E. I.; MOSER, M.-B. Path integration and the neural basis of the'cognitive map'. **Nature Reviews Neuroscience**, Nature Publishing Group, v. 7, n. 8, p. 663, 2006.

MENEZES, M. C.; FREITAS, E. P. de; CHENG, S.; OLIVEIRA, A. C. M. de; RIBEIRO, P. R. de A. A neuro-inspired approach to solve a simultaneous location and mapping task using shared information in multiple robots systems. In: IEEE. **2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)**. [S.l.], 2018. p. 1753–1758.

MILFORD, M.; PRASSER, D.; WYETH, G. Experience mapping: producing spatially continuous environment representations using ratslam. In: AUSTRALIAN ROBOTICS AND AUTOMATION ASSOCIATION INC. **Proceedings of Australasian Conference on Robotics and Automation 2005**. [S.l.], 2005.

MILFORD, M.; WYETH, G. Persistent navigation and mapping using a biologically inspired slam system. **The International Journal of Robotics Research**, SAGE Publications Sage UK: London, England, v. 29, n. 9, p. 1131–1153, 2010.

MILFORD, M.; WYETH, G.; PRASSER, D. Ratslam on the edge: revealing a coherent representation from an overloaded rat brain. In: IEEE. **Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on**. [S.l.], 2006. p. 4060–4065.

MILFORD, M. J.; WILES, J.; WYETH, G. F. Solving navigational uncertainty using grid cells on robots. **PLoS computational biology**, Public Library of Science, v. 6, n. 11, p. e1000995, 2010.

MILFORD, M. J.; WYETH, G. F. Mapping a suburb with a single camera using a biologically inspired slam system. **IEEE Transactions on Robotics**, v. 24, n. 5, p. 1038–1053, Oct 2008. ISSN 1552-3098.

MILFORD, M. J.; WYETH, G. F.; PRASSER, D. Ratslam: a hippocampal model for simultaneous localization and mapping. In: IEEE. **Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on**. [S.l.], 2004. v. 1, p. 403–408.

MOSER, E. I.; KROPFF, E.; MOSER, M.-B. Place cells, grid cells, and the brain's spatial representation system. **Annu. Rev. Neurosci.**, Annual Reviews, v. 31, p. 69–89, 2008.

MOURIKIS, A. I.; ROUMELIOTIS, S. I. Predicting the performance of cooperative simultaneous localization and mapping (c-slam). **The International Journal of Robotics Research**, Sage Publications Sage CA: Thousand Oaks, CA, v. 25, n. 12, p. 1273–1286, 2006.

O'KEEFE, J. Place units in the hippocampus of the freely moving rat. **Experimental neurology**, Elsevier, v. 51, n. 1, p. 78–109, 1976.

O'KEEFE, J.; BURGESS, N.; DONNETT, J. G.; JEFFERY, K. J.; MAGUIRE, E. A. Place cells, navigational accuracy, and the human hippocampus. **Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences**, The Royal Society, v. 353, n. 1373, p. 1333–1340, 1998.

O'KEEFE, J.; DOSTROVSKY, J. The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. **Brain research**, Elsevier Science, 1971.

PALMIERI, N.; RANGO, F. D.; YANG, X. S.; MARANO, S. Multi-robot cooperative tasks using combined nature-inspired techniques. In: IEEE. **Computational Intelligence (IJCCI), 2015 7th International Joint Conference on**. [S.l.], 2015. v. 1, p. 74–82.

PALMIERI, N.; YANG, X.-S.; RANGO, F. D.; SANTAMARIA, A. F. Self-adaptive decision-making mechanisms to balance the execution of multiple tasks for a multi-robots team. **Neurocomputing**, Elsevier, v. 306, p. 17–36, 2018.

PRASSER, D.; MILFORD, M.; WYETH, G. Outdoor simultaneous localisation and mapping using ratslam. In: CORKE, P.; SUKKARIAH, S. (Ed.). **Field and Service Robotics**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. p. 143–154. ISBN 978-3-540-33453-8.

QUIGLEY, M.; CONLEY, K.; GERKEY, B.; FAUST, J.; FOOTE, T.; LEIBS, J.; WHEELER, R.; NG, A. Y. Ros: an open-source robot operating system. In: KOBE, JAPAN. **ICRA workshop on open source software**. [S.l.], 2009. v. 3, n. 3.2, p. 5.

RANGO, F. D.; PALMIERI, N.; YANG, X. S.; MARANO, S. Bio-inspired exploring and recruiting tasks in a team of distributed robots over mined regions. In: IEEE. **Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2015 International Symposium on**. [S.l.], 2015. p. 1–8.

RANGO, F. D.; PALMIERI, N.; YANG, X.-S.; MARANO, S. Swarm robotics in wireless distributed protocol design for coordinating robots involved in cooperative tasks. **Soft Computing**, v. 22, n. 13, p. 4251–4266, Jul 2018. ISSN 1433-7479. Disponível em: <https://doi.org/10.1007/s00500-017-2819-9>.

RANJBAR-SAHRAEI, B.; TUYLS, K.; CALISKANELLI, I.; BROEKER, B.; CLAES, D.; ALERS, S.; WEISS, G. 13 - bio-inspired multi-robot systems. In: NGO, T. D. (Ed.). **Biomimetic Technologies**. [S.l.]: Woodhead Publishing, 2015, (Woodhead Publishing Series in Electronic and Optical Materials). p. 273 – 299. ISBN 978-0-08-100249-0.

SAEEDI, S.; TRENTINI, M.; SETO, M.; LI, H. Multiple-robot simultaneous localization and mapping: A review. **Journal of Field Robotics**, Wiley Online Library, v. 33, n. 1, p. 3–46, 2016.

SHARMA, S.; SUR, C.; SHUKLA, A.; TIWARI, R. Cbdf based cooperative multi robot target searching and tracking using ba. In: **Computational Intelligence in Data Mining-Volume 3**. [S.l.]: Springer, 2015. p. 373–384.

SILVA, G.; COSTA, J.; MAGALHÃES, T.; REIS, L. P. Cyberrescue: A pheromone approach to multi-agent rescue simulations. In: IEEE. **Information Systems and Technologies (CISTI), 2010 5th Iberian Conference on**. [S.l.], 2010. p. 1–6.

SÜNDERHAUF, N.; PROTZEL, P. Beyond ratslam: Improvements to a biologically inspired slam system. In: IEEE. **Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on**. [S.l.], 2010. p. 1–8.

______. Learning from nature: biologically inspired robot navigation and slam—a review. **KI-Künstliche Intelligenz**, Springer, v. 24, n. 3, p. 215–221, 2010.

TAUBE, J. S.; MULLER, R. U.; RANCK, J. B. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. **Journal of Neuroscience**, Soc Neuroscience, v. 10, n. 2, p. 420–435, 1990.

THRUN, S.; LEONARD, J. J. Simultaneous localization and mapping. **Springer handbook of robotics**, Springer, p. 871–889, 2008.

XBOT. Apostila de software do robodeck versão 1.1. [S.l.], 2011.

ZENO, P. J.; PATEL, S.; SOBH, T. M. Review of neurobiologically based mobile robot navigation system research performed since 2000. **Journal of Robotics**, Hindawi, v. 2016, 2016.