RoboCupRescue - Simulation League Team RescueRobots Freiburg (Germany)

Alexander Kleiner and Vittorio Amos Ziparo

Institut für Informatik
Foundations of AI
Universität Freiburg,
79110 Freiburg, Germany

http://www.informatik.uni-freiburg.de/rescue/∼virtual

Abstract. This paper describes the approach of the *RescueRobots Freiburg Virtual League* team. Our simulated robots are based on the two real robot types *Lurker*, a robot capable of climbing stairs and random stepfield, and *Zerg*, a lightweight and agile robot, capable of autonomously distributing RFID tags.
Our approach covers a novel method for RFID-Technology based SLAM and exploration, allowing the fast and efficient coordination of a team of robots. Furthermore we utilize *Petri* nets for team coordination.

1 Introduction

This paper describes the approach of the *RescueRobots Freiburg Virtual League* team. In general, our research focuses on the implementation of a fully autonomous team of robots that quickly explores a large terrain while mapping the environment. The simulated robots are based on the two real robot types *Lurker*, a robot capable of climbing stairs and random stepfield, and *Zerg*, a lightweight and agile robot, capable of autonomously distributing RFID tags.

Our approach covers a novel method for RFID Technology-based SLAM and exploration, allowing the fast and efficient coordination of a team of robots. The motivation behind RFID-Technology based SLAM and exploration is the simplification of the 2D mapping problem by RFID tags, which the robots autonomously distribute with a tag-deploy-device. RFID tags provide a world-wide unique number that can be read from distances up to one meter. The detection of these tags and thus the unique identification of locations is significantly computationally cheaper and less erroneous than identifying locations from camera images and range data [1].

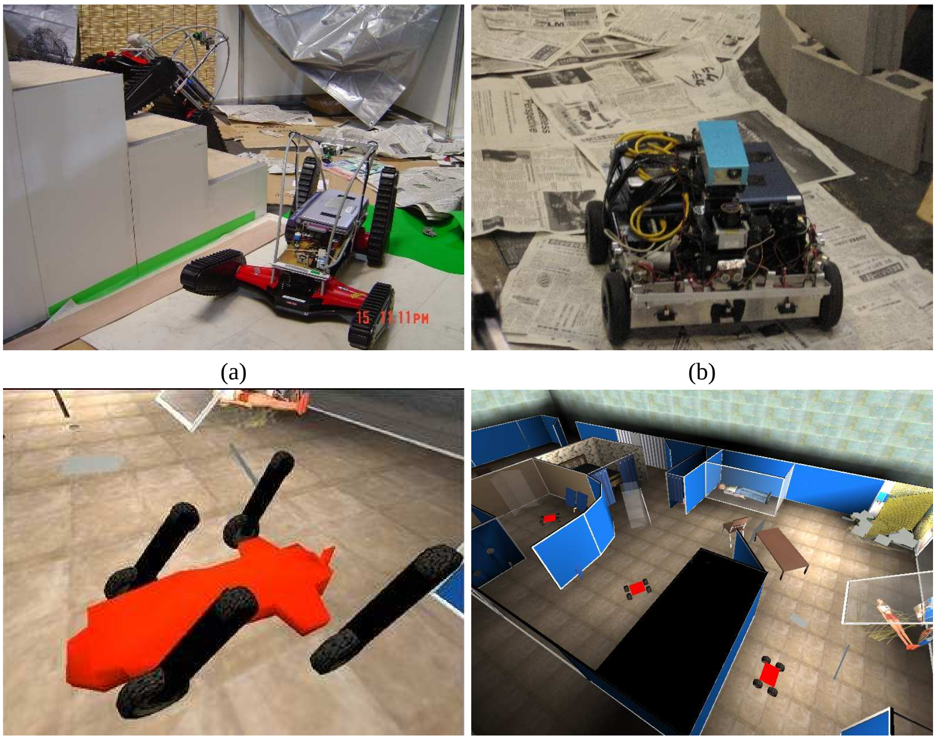
RFID-Technology based SLAM and exploration has advantages for Urban Search and Rescue (USAR): The system generates from RFID tags a topological map, which can be augmented with structural and victim-specific information. If human task forces are also equipped with RFID readers, they can directly localize themselves within this map, rather than locating themselves in a 2D or 3D metric map. Travel routes to victims can directly be passed to them as complete plans that consist of RFID tag locations and

[1] Note that even for humans the unique identification of a location is hard, when, for example, exploring a large office building or a collapsed building structure.

directions to follow. In fact, tags can be considered as signposts since the topological map provides for each tag the direction to the next tag. Furthermore, it is possible to store data, e.g. concerning nearby rooms or victims, directly in the tags. This information can then be utilized by other teams that are out of communication range.

The idea of labeling locations with information that is important to the rescue task has already be applied in practice. During the disaster relief in New Orleans in 2005, rescue task forces marked buildings with information concerning, for example, hazardous materials or victims inside the buildings. Our RFID-Technology based marking of locations is a straight forward extension of this concept.

Furthermore we utilize *Petri* nets for team coordination.

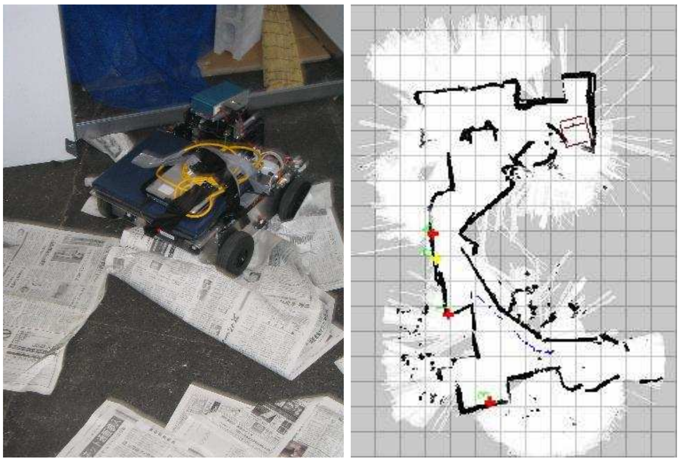


(a) (b) (c) (d)

Fig. 1. Robots built by our team: (a) The real *Lurker* robot and (b) the real *Zerg* robot at the RoboCup competition in Osaka. (c) The simulated *Lurker* robot and (b) the simulated *Zerg* robot. Picture (a) was taken by Adam Jacoff.

2 RFID Technology-based SLAM

The *RescueRobots Freiburg* real robot team successfully performed SLAM during the final of the *Best in Class autonomy* competition at RoboCup 2005 in Osaka. The map shown in figure 2 (b) was autonomously generated by the system, i.e. directly printed out after the mission without any manual adjustment of the operator. Our overall system



(a) (b)

Fig. 2. *Zerg* robot during the final of the *Best in Class autonomy* competition at RoboCupRescue 2005 in Osaka: (a) slipping on newspapers and (b) the autonomously generated map. Red crosses mark locations of victims which have been found by the robot.

for SLAM is based on three levels, which are: *Slippage-sensitive odometry*, *Scanmatching*, and *RFID-based localization*. From these three levels, the latter two are applied within the *Virtual Robot competition*.

We tackle the “Closing The Loop” problem by actively distributing unique RFID tags in the environment, i.e. placing them automatically on the ground, and by utilizing the tag correspondences found on the robot’s trajectory for calculating globally consistent maps after the method introduced by Lu and Milios [1]. This method requires reliable estimates of the local displacement between two RFID tags. Therefore, a Kalman filter is utilized, which estimates the robot’s pose from both scan matching and odometry-based dead reckoning.

Generally, the robot’s pose can be modeled by a Gaussian distribution $N(l, \Sigma_l)$, where $l = (\hat{x}, \hat{y}, \hat{\theta})^T$ is the mean and Σ_l a 3×3 covariance matrix, expressing uncertainty of the pose [2]. Given the measurement of the robot’s motion by the normal distribution $N(u, \Sigma_u)$, where $u = (d, \alpha)$ is the input of traveled distance d and angle α, respectively, and Σ_u a 2×2 covariance matrix expressing odometry errors, the robot’s pose at time t can be updated as follows:

$$l_t = F(l_{t-1}, d, \alpha) = \begin{pmatrix} \hat{x}_{t-1} + \cos(\hat{\theta}_{t-1})d \\ \hat{y}_{t-1} + \sin(\hat{\theta}_{t-1})d \\ \hat{\theta}_{t-1} + \alpha \end{pmatrix}, \tag{1}$$

$$\Sigma_{l_t} = \nabla F_l \Sigma_{l_{t-1}} \nabla F_l^T + \nabla F_u \Sigma_{u_{t-1}} \nabla F_u^T, \tag{2}$$

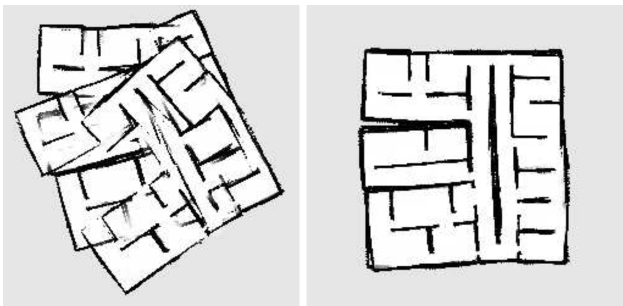
where F describes the update formula, and ∇F_l and ∇F_u are partial matrices of its Jacobian ∇F.

Suppose the robot distributes n RFID tags at unknown locations $l_1, l_2, ..., l_n$, with distance $d_{ij} = (\Delta x_{ij}, \Delta y_{ij}, \Delta\theta_{ij})$ between l_i and l_j. In order to determine the estimated distance $\hat{d}_{ij}$ with corresponding covariance matrix Σ_{ij} between l_i and l_j, the previously described Kalman filter is utilized. If the robot passes a tag l_i, we reset the Kalman Filter in order to estimate the *relative* distance $\hat{d}_{ij}$ to subsequent tag l_j on the robot's trajectory.

Our goal is it to estimate locations l_i that best explain the measured distances $\hat{d}_{ij}$ and covariances Σ_{ij}. This can be achieved with the maximum likelihood concept by minimizing the following Mahalanobis-distance:

$$W = \sum_{ij} \left(d_{ij} - \hat{d}_{ij} \right)^T \Sigma_{ij}^{-1} \left(d_{ij} - \hat{d}_{ij} \right), \tag{3}$$

where the summation goes over all measured distances and d_{ij} is the true distance between l_i and l_j. Note if we assume the robot's orientation to be measured by the IMU (whose error does not accumulate), we do not need to consider the orientation θ within the d_{ij} in Equation 3, and hence the optimization problem can be solved linearly by calculating $d_{ij} = l_i - l_j$. However, if we also want to improve the estimate of the orientation, the d_{ij} have to be linearized. It can easily be shown that the optimization problem in Equation 3 can be solved as long as the covariances Σ_{ij} are invertible [1]. For distributing the tags in the environment, we constructed a special aperture which is also simulated on the virtual *Zerg* robot.



(a) (b)

Fig. 3. Result from applying the non-linear mapper to data generated in the simulation. (a) Map with odometry noise and (b) the corrected map.

3 RFID Technology-based exploration

Efficiency of multi-robot exploration is usually measured by the ratio between the explored area and the distance traveled by the robots [5]. This efficiency can only be maximized if robots know about the past and future exploration targets of the other robots. The proposed method enables an exchange of this information via programmable RFID tags. Note that due to unconstrained communication in the *Virtual Competition*, information concerning RFID tags can directly be communicated by the robots without passing them.

We assume that a *single* robot explores the environment, based on the concept of "frontier cell" exploration [4]. A cell is considered as frontier cell if it has already been explored but also neighbors an unexplored cell. Each robot maintains a set of frontier cells with respect to its observations, e.g. by removing cells coming into the field of view of its sensors and by adding cells that are at the border, respectively. Frontier cells are generated by integrating laser range finder readings into an occupancy grid. In our approach, we restrict the size of this grid to the local vicinity of the robot.

The basic idea of RFID-based exploration is to leave behind information via RFID tags, which helps other robots to reduce the overlap of exploration targets. Therefore, we store on RFID tags the relative locations of *visited cells* $V_{tag} = (\Delta v_1, \Delta v_2, ..., \Delta v_m)$, i.e. cells that have been visited by other robots. Tags are intended to provide information for a local area of the exploration space, thus their influence radius, i.e. maximal distance of relative locations, is limited by the distance τ. Robots subsequently synchronize the data related to a tag with their list of visited locations within the range τ of the tag. We assume that the robots IMU is based on a compass and thus the local coordinate frames of the robots are equally aligned to magnetic north.

Each robot maintains a collection V_r containing time-stamped locations $l_t = (x, y)$. During each cycle t, the robot adds its current pose l_t. If the robot passes a RFID tag, V_r and V_{tag} are synchronized after Algorithm 1. Furthermore, we maintain for each RFID

Algorithm 1 Synchronization of V_r and V_{tag}

```
for all Δv_i ∈ V_tag do
    add absolute location (Δv_i + l_0) to V_r
end for
for all v_j ∈ V_r do
    if ‖ v_j − l_0 ‖ < τ then
        add relative location (v_j − l_0) to V_tag
    end if
end for
```

tag location a local evidence grid that integrates the observations from the victim sensor. Each observation is updated according to the robots pose and the sensors field of view. From the victim evidence grid a second set of frontier cells is calculated. According to the two sets of frontier cells and the locations visited by other robots, exploration targets are selected with different priority, whereas infrequently explored areas are preferred. Figure 4 shows some results of the RFID Technology-based exploration. Note that these results where obtained within a free-space exploration, i.e. within an arena without victims.

4 Coordination

In order to successfully explore and navigate in the arena robots need to coordinate. In particular, we rely on a centralized approach in which an agent is reponsable for collecting relevant information and producing a multiagent synchronized plan.

The coordinator agent will build and maintain a dynamic model of the world based on Petri nets [3]. The current knowledge of the environment and the state of the multia-

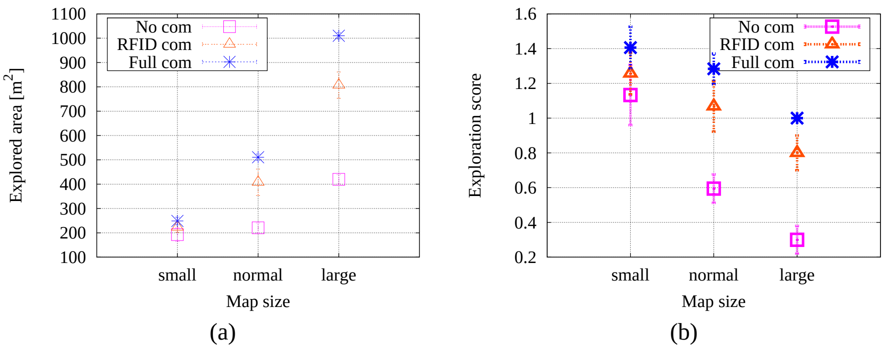


Fig. 4. Averaged results from various exploration runs in the simulation. (a) and (b) the exploration score, i.e. area divided by travelled distance.

gent system will thus be represented as a Petri net $< P, T, F, W, M_0 >$. The model will be dynamically updated as information is gathered and will be used to compute moves for each agent which guarantee a safe multiagent path planning.

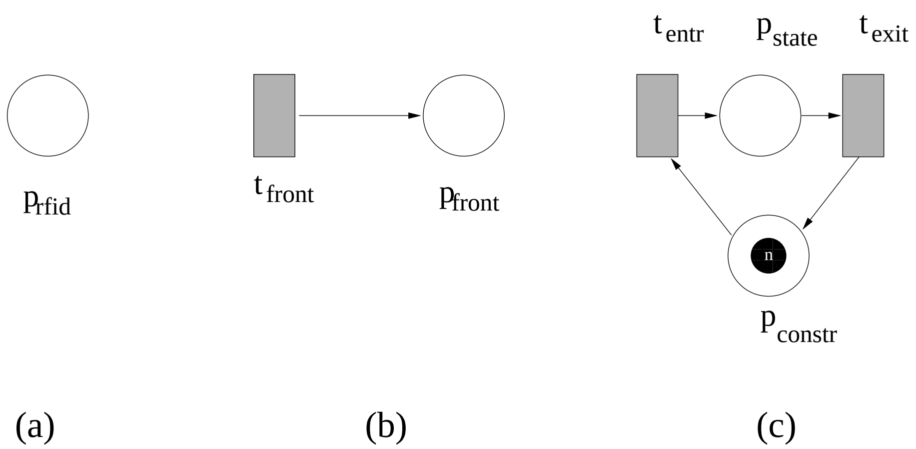


Fig. 5. Basic structures: (a) the *RFID location* structure (b) the *Frontier* structure and (c) the *Passage* structure.

The following structures $< P, T, F >$ and the their possible combinations define the syntactic structure of our model.

1. *RFID Location*: This structure represents a unique RFID location. Figure 5(a) shows a graphical representation. Formally: $< \{p_{rfid_i}\}, \{\emptyset\}, \{\emptyset\} >$.
2. *Frontier*: This structure represents the open frontiers in the vicinity of an RFID location. Figure 5(b) shows a graphical representation of the structure. Formally: $< \{p_{front}\}, \{t_{front}\}, \{(t_{front}, p_{front})\} >$.

3. *Passage*: This structure represents a connection between RFID locations. Figure 5(c) shows a graphical representation. The marking of p_{constr} denotes the maximum amount of robots allowed in the passage simultaneously. More formally a passage is : $< \{p_{state}, p_{constr}\}, \{t_{entr}, t_{exit}\}, \{(t_{entr}, p_{state}), (p_{state}, t_{exit}), (t_{exit}, p_{constr}), (p_{constr}, t_{entr})\} >$.

We combine these structures to obtain new models:

1. A RFID Location $< P_l, T_l, F_l >$ can be combined with a frontier $< P_f, T_f, F_f >$ resulting in $< P_l \cup P_f, T_l \cup T_f, F_l \cup F_f \cup \{(p_{rfid}, t_{front})\} >$
2. A RFID Location $< P_l, T_l, F_l >$ can be combined with a passage $< P_p, T_p, F_p >$ resulting in $< P_l \cup P_p, T_l \cup T_p, F_l \cup F_p \cup \{(p_{rfid}, t_{entr})\} >$
3. A passage $< P_p, T_p, F_p >$ can be combined with a RFID Location $< P_l, T_l, F_l >$ resulting in $< P_l \cup P_p, T_l \cup T_p, F_l \cup F_p \cup \{(t_{exit}, p_{rfid})\} >$

Algorithm 2 Coordination Agent

```
while True do
  for all Task ∈ AccomplishedTasks do
    if Task.isInitialLocation() then
      Model.addRFIDLocation(Task.RFID)
    else
      if Task.isFrontier() then
        Model.addRFIDLocation(Task.RFID)
        Model.removeFrontier(Task)
        Model.combine(Task.PrevRFID, Task.Passage)
      end if
    end if
    Model.updateMarking(Task)
    for all Neighbor ∈ Task.FrontiersList() do
      if Neighbor ∉ ExploredList then
        Model.combine(Task.RFID, Neighbor)
      end if
    end for
  end for
  Plan = calculatePlan(Model)
  assignGoals(Plan.nextStep)
  Model.updateMarking(Plan)
end while
```

Assuming robots will deploy an RFID at their starting location the Algorithm 2 correctly maintains the model and assigns tasks to agents. In particular, the model is build as follows:

1. Whenever an RFID is deployed by an agent a new place is added to the graph. The marking of this node represents the number of agents in the proximity of the RFID. In particular, an agent will be associated to the nearest RFID within those in a distance of τ (Section 3).

2. The RFID locations are combined with the open frontiers perceived at the by each agent.
3. When moving from a location to a frontier robots will identify *passages*. Passages cannot be longer than τ which is the maximum distance between two RFID locations. The former structure is thus characterized by a distance $d < \tau$ and a constraint on how many agents can simultaneously travel in the passage (i.e. the marking of the constraint place p_{constr}).

Given the current model we can search for the sequence of state transitions that maximize the overall performance: i.e. find a multiagent plan that maximizes the number of frontier places with marking one and minimizes the travelled distance of the agent who travels the most.

At each timestep, given the current model we can assign a RFID location or a frontier neighboring each robot. The overall result of this continuous planning process will be a synchronized multiagent path plan which guarantees that the assignments are safe (in the sense that passage capacities are not exceeded) and optimal according to the current knowledge.

5 Conclusion

Our approach offers a solution to the problem of the deployment of a large group of robots while utilizing as less as necessary computational resources. This is carried out by the decomposition of the generally computational hard problem of SLAM, exploration and team coordination into two levels: Firstly, the grid based level, which is locally restricted to the close vicinity of a robot or a RFID tag. Secondly, the topological level, which is less fine grained than the local level, however, allows to efficiently compute globally consistent solutions. We belief that this decomposition leads to an efficient solution to the problem of multi-robot search and rescue.

References

1. F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Auton. Robots*, 4:333–349, 1997.
2. P. S. Maybeck. *Stochastic models, estimation, and control*, volume 141 of *Mathematics in Science and Engineering*. 1979.
3. Tadao Murata. Petri nets: Properties, analysis and applications. In *Proceedings of the IEEE*, pages 541–580, April 1989. NewsletterInfo: 33Published as Proceedings of the IEEE, volume 77, number 4.
4. B. Yamauchi. A frontier-based approach for autonomous exploration. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97)*, 1997.
5. R. Zlot, A. Stentz, M. Dias, and S. Thayer. Multi-robot exploration controlled by a market economy. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002.