REGION-BASED HEURISTICS FOR AN ITERATIVE PARTITIONING PROBLEM IN MULTIAGENT SYSTEMS

Thomas Kemmerich
International Graduate School Dynamic Intelligent Systems, University of Paderborn, 33095 Paderborn, Germany

Hans Kleine Büning
Department of Computer Science, University of Paderborn, 33095 Paderborn, Germany

Keywords: Region-based heuristic, Partitioning, Multi-objective optimization, Multiagent system, Coordination, External storage media.

Abstract: Load balancing or access point selection in wireless networks both are problems where a large set of particles repeatedly has to be partitioned on another set of objects. In general this partitioning problem involves multiple contrary objectives. Due to the large number of particles a decentralized approach should be favored. In this work, such an iterative multi-objective optimization problem is modeled as multiagent system. We propose a local solution technique based on regions and some special coordination media. Agents select target objects based on the region they are in. Different region types are considered and a local heuristic is developed. We show the general potential of regions and experimentally analyze different approaches. All approaches are able to provide high quality solutions.

1 INTRODUCTION

Repeated partitioning of some elements onto another set of special elements is a frequently occurring problem in a variety of different real systems. For instance when multiple users have to be allocated to some servers. Another example can be found in Wireless LANs, where mobile devices have to select an access point (AP) from a set of reachable APs. In the latter system, recent AP selection protocols still are unable to solve the problem appropriately in all settings (Yen et al., 2009). The used protocols basically make a mobile device to independently select the AP with the strongest signal to noise ratio. Since bandwidth at an AP has to be shared among all participants, one can easily imagine settings where such simple protocols will result in unbalanced assignments of devices to access points, esp. if large numbers of devices are involved. Several works deal with this issue in WLANs, e.g. (Yen et al., 2009), or (Kasbekar et al., 2006).[1]

In this work we consider such a repeated or iterative partitioning problem that involves multiple optimization criteria. We model it as dynamic multiagent system (MAS) where agents change positions over time and agents as well as mechanisms should be as simple as possible. Based on local information, agents have to partition themselves onto a set of target objects. The created partitionings are supposed to be optimized against some contrary objectives.

Up to now, even no efficient *central* approach to solve this general problem in scenarios with more than two targets and an arbitrary number of agents is known (Goebels, 2007). We conjecture that the problem is at least NP-hard. Hence, efficient heuristics have to be developed. As we are interested in local algorithms in that field, we propose to approach the problem using regions that are defined based on target positions. Information about these regions will be used locally by agents to determine a target assignment. At the same time these local decisions should lead to good partitionings according to global objectives. Besides our focus on local approaches, we are also interested in building simple algorithms which only involve very simple knowledge structures and reasoning processes. The question is how simple both can become on the one hand while on the other hand they still should enable agents to find good solutions.

Section 2 briefly deals with related work. Sec-

[1]Note that the presented approaches may not directly be transfered to real systems as load balancing has to take into account traffic patterns, too. However, our results could be considered in the design of novel AP selection mechanisms.

Kemmerich T. and Kleine Büning H..
REGION-BASED HEURISTICS FOR AN ITERATIVE PARTITIONING PROBLEM IN MULTIAGENT SYSTEMS.
DOI: 10.5220/0003151202000205
In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (ICAART-2011), pages 200-205
ISBN: 978-989-8425-41-6

tion 3 then presents a detailed model of the problem. Next, we proceed stepwise to develop a local heuristic to solve the given problem. We will first define the considered *Target-Regions* in Section 4. These regions are defined based on target positions in static scenarios and build the basis of the heuristics developed in this paper. An experimental[2] investigation of an hypothesis on the solution quality of approaches that use Target-Regions is given. As Target-Regions represent a good mean to solve the considered problem but at the same time are hard to calculate, we have to approximate them for general scenarios. Therefore, we propose an approximation technique in Section 5 that will be evaluated, too. Based on these insights we finally develop the local approach. It will use so-called *storage media* which can be considered as coordination objects as they are used by agents to externally store information about "good" target assignments. The experimental results show that this local approach is able to find high quality solutions in general settings. In the end, we conclude this work in Section 6.

2 RELATED WORK

The problem underlying the iterative partitioning task considered in this work is based on the Online Partitioning Problem (Goebels, 2007). In (Ducatelle et al., 2009), the authors solve a similar problem using a communication-based and a reactive approach.

The iterative problem considered in this work demands an assignment of agents to targets and thereby in a sense a formation of groups. In contrast to clustering or (iterative) graph partitioning (Fjällström, 1998), where elements are clustered according to some metrics, an additional constraint applies in our problem as targets have to be in different groups.

In the area of RoboCup Rescue, task allocation algorithms are investigated, too. In (Sedaghat et al., 2006), a simple partitioning technique that divides a map in regular regions in combination with an auction-based mechanism is shown to be an effective mean to solve their task allocation problem.

The local heuristic developed in this work is based on a framework that incorporates external storage media in the reasoning process of a capacity-constrained multiagent systems (MAS) (Kemmerich and Kleine Büning, 2010a). Knowledge of agents is stored on passive external storage media (SM) that are located within the environment. Hence, storage media

[2]Source code, evaluation scripts, and configuration files of all experiments conducted in this work are available for download at http://www.upb.de/cs/ag-klbue/en/staff/kemmerich/icaart10.tgz

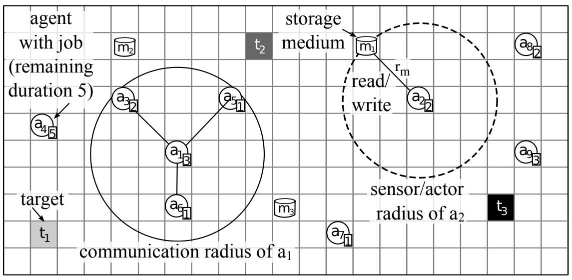


Figure 1: MAS with agents, jobs, targets, and external storage media.

become a mean for the coordination of agents. Storage media are comparable to *coordination artifacts* as introduced in (Omicini et al., 2004). These artifacts are also meant to support coordination in MAS and finally resulted in the A&A meta-model (Omicini et al., 2008). In that context, environments as first class abstractions in MAS to support coordination, cooperation, and interaction between agents recently gain an increasing interest (Parunak and Weyns, 2007).

3 PROBLEM STATEMENT

Figure 1 illustrates the considered MAS which, besides using a grid environment, is basically in a line with (Kemmerich and Kleine Büning, 2010a). It consists of a set of agents $\mathcal{A} = \{a_1, \ldots, a_n\}$, a set of targets $\mathcal{T} = \{t_1, \ldots, t_m\}$, and a set $\mathcal{M} = \{m_1, ..., m_q\}$ of external storage media which will be used for coordination purposes later in this work. All entities are placed in a 2-dimensional Euclidean grid environment $\mathcal{E}$ as defined in Definition 1.

Definition 1 ($\mathcal{A}$,$\mathcal{T}$-Grid-Environment)
An $\mathcal{A}$,$\mathcal{T}$-Grid-Environment $\mathcal{E}$ is a rectangular grid environment containing $|\mathcal{A}|$ agents, $|\mathcal{T}|$ targets, and a set of storage media $\mathcal{M}$. Each object is located within one cell of the grid, i.e. no two objects intersect. Let $c = \text{size}(\mathcal{E})$ denote the total number of cells in the grid. Then we call an $\mathcal{A}$,$\mathcal{T}$-Grid-Environment full, *if and only if $c = |\mathcal{A}| + |\mathcal{T}|$, and* sparse *otherwise.*

Each agent works on a job located in the environment. Each job $j \in \mathcal{J}$ requires $ct(j) \in [\min_t, \max_t]$ time steps and is executed till completion. Job positions and durations are chosen uniformly at random. If a job is done, the environment assigns the agent to a newly created random job. Accordingly, this procedure leads to dynamically changing agent positions while target and storage media positions are fixed. Agents interact locally with storage media by reading, writing, and deleting information items. The model further limits the internal memory capacity of agents and storage media using a ring buffer. This implies that information might be overridden. The neighborhood $\mathcal{N}_a$ of agent a consists of up to k agents that are nearest to a within a given communication radius.

Let $p : \mathcal{A} \to \mathcal{T}$ be a total function that maps elements from $\mathcal{A}$ to elements from $\mathcal{T}$. Then a *partitioning* of two non-empty sets $\mathcal{A}$ and $\mathcal{T}$ is defined as a multiset $\mathcal{S}_{\mathcal{A},\mathcal{T}} = \{S_1, S_2, \ldots, S_{|\mathcal{T}|}\}$ having $S_i = \{a \in \mathcal{A} \mid p(a) = t_i\}$.

The considered problem demands a repeated partitioning of agents to targets. The goal is to find *good* partitionings in each iteration using only local information that optimize three contrary objectives:

1. create a **uniform distribution**, i.e. assign approximately $\frac{|\mathcal{A}|}{|\mathcal{T}|}$ agents to each target
2. minimize the **distance sum** between agents and selected targets
3. minimize the **costs** that are produced according to a cost model

To measure the quality of a partitioning established in iteration ℓ, we use a function $f : \mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}} \to [0,1]$ that is calculated from a global perspective. In this work, Equation 1 realizes this function and calculates the weighted sum of the first two normalized objective values. Weights α and β in general must be greater or equal to zero and sum to one. If not stated otherwise, we will use $\alpha = \beta = \frac{1}{2}$. Note that criteria three (*costs*) is not considered in this formula as it does not directly influence the pure partitioning quality. For a discussion on how costs could be modeled and for a brief cost analysis, we refer to the extended version of this paper (Kemmerich and Kleine Büning, 2010b).

$$f(\mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}}) = \alpha \cdot \left(\frac{\prod\limits_{S_i \in \mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}}} |S_i|}{\left(\frac{|\mathcal{A}|}{|\mathcal{T}|}\right)^{|\mathcal{T}|}} \right) + \beta \cdot \left(\frac{\sum\limits_{a \in \mathcal{A}} \delta(a, \tau(a))}{\sum\limits_{a \in \mathcal{A}} \delta(a, p(a))} \right) \tag{1}$$

The first term of Equation 1 represents the distribution objective value derived from a partitioning $\mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}}$. It is normalized against an optimal uniform distribution. The second term normalizes the distance sum of $\mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}}$ against the minimal possible distance sum. Here, $\tau : \mathcal{A} \to \mathcal{T}$ is used which returns the nearest target of an agent (ties are broken by selecting the target t_i with the lowest index i). The Euclidean distance between an agent and a target is calculated by $\delta : \mathcal{A} \times \mathcal{T} \to \mathbb{R}^+$. It is particularly noticeable that agents cannot calculate partitioning qualities on their own due to partial observability of the environment.

The overall goal of the agents then is to maximize the average partitioning quality over some iterations k as expressed by

$$q = \frac{1}{k} \sum_{\ell=1}^{k} f(\mathcal{S}^{\ell}_{\mathcal{A},\mathcal{T}}). \tag{2}$$

4 TARGET-REGIONS

We now propose the basic mechanism upon which the final local heuristic will be based on. Therefore, we first define the term *Target-Regions*:

Definition 2 (Target-Region)
A Target-Region $\mathrm{TR}(t)$ *for any target t in a* full *$\mathcal{A}$-$\mathcal{T}$-Grid-Environment is defined by a set of cells C_t. The set consists of target t's cell and all cells of agents that, in an* optimal solution, *are assigned to t.*

Based on this definition, the basic idea is very simple: given Target-Regions and a *full* environment, agents create partitionings by just selecting the target of the region they are located in. The corresponding algorithm then reads as follows:

Algorithm 1: Executed by each agent $a \in \mathcal{A}$.

```
Determine current agent position pos(a)
Determine Target-Region TR(t) at pos(a)
Assign agent a to target t
```

Obviously, the decision making itself cannot be simpler besides having an build-in oracle that makes the agents know the *best* target to select at any time. Also the knowledge structure is very simple as it is only composed of one position and one target information item. Hence this idea fits well to our intentions of building simple algorithms.

In addition, this simple approach will — by definition of Target-Regions and since target positions are fixed — result in optimal solutions in full environments. The reasons therefore are that i) agent positions in full environments can also be considered to be fixed and ii) that agents are anonymous in the quality function defined in Equation 1. Accordingly, such a full scenario reduces to a static one, which means that TR provide optimal partitionings in each iteration and thus also result in an optimal average partitioning quality.

Although these regions were calculated for a *full* and thus static scenario, we also propose to solve the general iterative partitioning problem, which usually involves less agents, with the help of these static regions. Since we then work in so-called *sparse* environments, Target-Regions by definition may not provide optimal solutions in all iterations.

At this point, two questions arise: the first question concerns the potential of static Target-Regions as a mean to solve the iterative problem in *sparse* environments. The second question is about the complexity of determining Target-Regions in static scenarios.

We investigate question one using Hypothesis 1.

Hypothesis 1. Let agents in an $\mathcal{A}$-$\mathcal{T}$-Grid-Environment repeatedly create partitionings based on Target-Regions. Then the resulting average partitioning quality according to Equation 2 is expected to be

1. high if agents are distributed uniformly, and
2. low if agents are distributed according to a normal distribution.

The solution quality further depends on the ratios between the number of agents, targets, and the size of the environment.

Note that if agents are normally distributed then they are concentrated within a certain part of the environment. This may lead to situations where no or only a small number of agents is located in some Target-Regions. Accordingly, bad distribution values and thus bad overall partitioning qualities result. Hence, in the remainder of this work we concentrate on uniformly distributed agents. In addition, the results of an empirical analysis fully support Hypothesis 1. The corresponding experiments compare optimal solutions for settings with two targets to those obtained using Algorithm 1 and static Target-Regions in sparse environments.[3] Details can be found in the extended paper (Kemmerich and Kleine Büning, 2010b).

The second question concerning the complexity of calculating Target-Regions is not yet answered. As by definition, Target-Regions are based on optimal solutions, their construction requires to optimally solve the partitioning problem. However, as already mentioned we conjecture that solving such static scenarios with more than two targets is at least NP-hard, since no efficient algorithm is known (Goebels, 2007). Thus, the problem of determining Target-Regions is conjectured to be at least NP-hard, too.

[3]We considered settings with two targets, because no polynomial-time algorithm that provably returns an optimal solution for settings with an arbitrary number of targets is known (Goebels, 2007). Accordingly, validating Hypothesis 1 for general settings is computationally intractable. However, we are aware of a central-instance polynomial-time algorithm for settings with two targets which we used in the evaluation.

5 APPROXIMATION OF TARGET-REGIONS

As the experiments conducted for validating Hypothesis 1 resulted in high quality solutions for uniformly distributed agents and because the construction of Target-Regions is assumed to be at least NP-hard, we propose to use approximated Target-Regions. The presented approximation is based on a local algorithm that is known as *Exchange Target Strategy* (ETS) (Goebels, 2007). Hence, we call the approximated regions *ETS-Target-Regions*.

5.1 ETS-Target-Regions

According to (Goebels, 2007), the *Exchange Target Strategy* (ETS) is a good mean to find high quality partitionings of agents to targets in settings with static positions. The basic idea of the ETS is as follows. Initially, agents are (randomly) assigned to targets. Then, agents repeatedly communicate assignment and distance information. They exchange target assignments with neighboring agents if this locally improves the distance objective. Thus, the distribution objective is fixed based on the initial assignment while the distance objective gradually improves until it converges. More details on ETS can be found in the extended version of the paper or in (Goebels, 2007).

Although ETS provides high quality solutions on average, worst cases leading to poor solutions or local optima can be constructed (Goebels, 2007). In addition, the costs produced by repeated information exchange may become relatively high.

To approximate Target-Regions, we propose to use the Exchange Target Strategy (ETS). We define the resulting regions in Definition 3.

Definition 3 (ETS-Target-Region)
An ETS-Target-Region *ETS-TR*(t) *for any target* t *in a full* $\mathcal{A}$-$\mathcal{T}$*-Grid-Environment is defined by a set of cells that consists of target* t*'s cell and all cells whose agents are assigned to* t *after the Exchange Target Strategy has converged.*

Note that ETS-Target-Regions (ETS-TR) in this work are those that have evolved after 2000 iterations of the ETS approach, as hand-made experiments indicated that this value was by far sufficient for convergence in all considered scenarios. Convergence in this context means that no further improvement of the overall solution quality was observed after 2000 iterations.

We developed an approach that first calculates ETS-Target-Regions for a *full* environment. The resulting ETS-TR then are mapped to the cells of the grid environment, i.e. each cell obtains an information about the region it belongs to. In a second phase, this initialized environment then can be used with the desired number of agents. Each agent simply selects the target that is identified by the information stored at the agent's current position.

Again, we simulated this approach and compared the results to optimal solutions for settings with two targets. We calculated an average error, which we defined as the average difference between the optimal solution value and the solution value obtained using the ETS-Target-Regions over all iterations. We found that ETS-TR with an *appropriate* ratio between the number of agents and the environment size is roughly 1% or less in seven out of nine simulated scenarios. Accordingly, we can conclude that ETS-TR are a good mean to approach the iterative partitioning problem, too. A more detailed description and the experimental evaluation can be found in the extended paper (Kemmerich and Kleine Büning, 2010b).

5.2 Sparse-ETS-Target-Region Algorithm

As indicated in Section 5.1, ETS-TR are a promising mean to solve our iterative partitioning task. However, it is impractical to consider a full $\mathcal{A}$-$\mathcal{T}$-Grid-Environment or to apply an approach that calculates ETS-Target-Regions to initialize the cells. In this section we thus introduce a local algorithm for sparse environments that approximates ETS-Target-Regions. Therefore, we use so-called storage media for coordination (Kemmerich and Kleine Büning, 2010a). Storage media (SM) are located at fixed positions in the environment and can be used by agents to store information externally.

The *Sparse ETS-Target-Region* (S-ETS-TR) approach can basically be divided into two phases. In the first phase, each agent locally executes the ETS algorithm for a fixed number of iterations $i_{\max}$. After each ETS iteration, every agent a stores its current target assignment combined with its current position $pos(a)$ on a storage media in its vicinity. Therefore, it must be guaranteed that each agent can always interact with a storage media. Then, the combination of all information stored on all storage media from a global perspective represents approximated ETS-Target-Regions, as sketched in Figure 2.

In the second phase agents stop to execute the ETS algorithm. Instead, they retrieve a target assignment from the nearest SM based on the information stored at the media. These information approximate ETS-Target-Regions. Depending on the parameters, there will be positions without target assignment information (white boxes in Figure 2). If this happens, a given position must be classified according to locally available information. Therefore, a SM executes a simple classification algorithm that uses a counting argument based on eight surrounding positions. The resulting classification basically corresponds to the most often selected target in the surrounding. It is returned and stored on the storage media for later use.

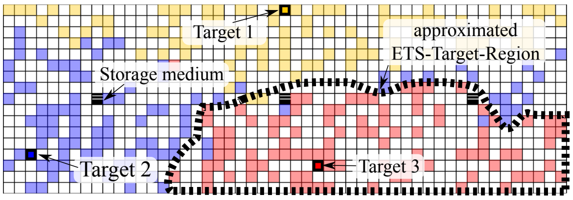


Figure 2: Exemplary target assignment information (colored boxes) stored on the storage media after some S-ETS-TR iterations.

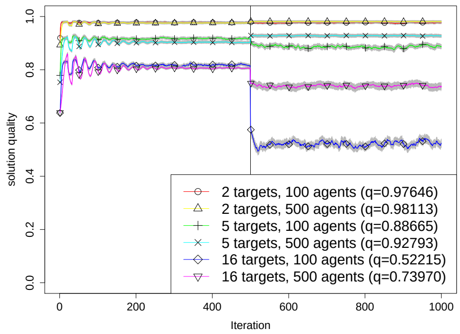
Algorithm 2 summarizes the S-ETS-TR approach. Details on the data structures and on the classification algorithm as well as on the experimental results are given in the extended paper.

Algorithm 2: Executed by each agent $a \in \mathcal{A}$.

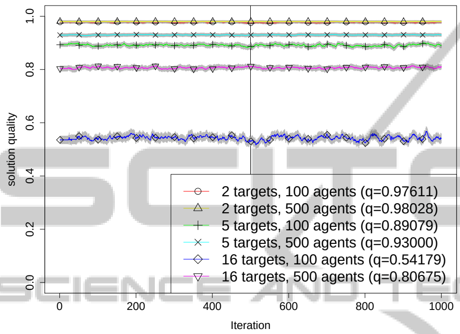
procedure SPARSE-ETS-TARGET-REGIONS
- $m \leftarrow$ nearest SM
- **if** current iteration $< i_{\max}$ **then** ▷ Phase 1
  - $t \leftarrow$ target assignment of an ETS iteration
  - $m.$STORE-INFO$(t, pos(a))$
- **else** ▷ Phase 2
  - $t \leftarrow m.$CLASSIFY-POSITION(pos(a))
- **if** t is set **then** assign agent to t
- **else** keep last target assignment

Figure 3 briefly summarize the latter results. It illustrates the process of both ETS-based approaches and shows 95% confidence intervals. The vertical line at iteration 500 marks the beginning of the second phase. It is particularly noticeable that phase one of S-ETS-TR may produce higher solution values as optimization is performed in each iteration by executing the ETS algorithm. Performance decreases in the second phase depending on the settings, esp. in settings with a higher target to agent ratio. However, with a lower ratio, the results in the second phase become even better (e.g. for 5 targets and 500 agents).

Comparing the average partitioning quality q, we observe that the local S-ETS-TR heuristic is able to produce high quality solutions that are on a level with the real ETS-TR approach from Section 5.1. This is particularly interesting as the very simple classification technique misclassifies between 5% and 23% of the cells compared to real ETS-Target-Regions.



(a) S-ETS-TR



(b) ETS-TR

Figure 3: Solution qualities with 95% confidence intervals.

The results of the ETS-TR approach in Figure 3(b) also illustrate the potential of ETS-based Target-Region approximations in settings with many targets. However, they are also affected by a dependency on the ratio between the number of agents and targets as already stated for Target-Regions in Hypothesis 1.

6 CONCLUSIONS

We constructed a local region-based heuristic to solve an iterative partition task in a multiagent system. The potential of the general idea and different region types were investigated experimentally. Our simulation results, although more simulations in different settings should be performed in the future, attribute high potential to region based heuristics. To improve solution qualities of the local approach, other classification techniques should be investigated to reach the potential that approximated regions can provide. Returning to our intension of building systems that are able to solve the iterative partitioning problem with very basic and simple information structures and reasoning processes, region-based approaches provide a promising mean and should be investigated further.

The extended version of this work (Kemmerich and Kleine Büning, 2010b) provides additional details on the approaches and their evaluation and proves some properties of optimal solutions and Target-Regions. We also present a formula to estimate solution qualities of the ETS-TR approach and discuss some cost issues concerning the local heuristic.

REFERENCES

Ducatelle, F., Förster, A., Di Caro, G., and Gambardella, L. (2009). New task allocation methods for robotic swarms. In *9th IEEE/RAS Conference on Autonomous Robot Systems and Competitions*.

Fjällström, P.-O. (1998). Algorithms for graph partitioning: A survey. *Linkping Electronic Articles in Computer and Information Science*, 3(10).

Goebels, A. (2007). *Agent Coordination Mechanisms for Solving a Partitioning Task*. Logos.

Kasbekar, G., Kuri, J., and Nuggehalli, P. (2006). Online association policies in IEEE 802.11 WLANs. In *4th International Symposium on Modeling and Optimization in Mobile, Ad-Hoc and Wireless Networks (WiOpt 2006)*, pages 11–20. IEEE.

Kemmerich, T. and Kleine Büning, H. (2010a). External coordination media in capacity-constrained multiagent systems. In *Proc. IEEE/WIC/ACM Intl. Joint Conf. on Web Intelligence and Intelligent Agent Technology (WI-IAT'10)*, pages 109–116. IEEE Computer Society.

Kemmerich, T. and Kleine Büning, H. (2010b). Region-based heuristics for an iterative partitioning problem in multiagent systems (extended version). Technical Report TR–RI–10–320, University of Paderborn.

Omicini, A., Ricci, A., and Viroli, M. (2008). Artifacts in the A&A meta-model for multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 17(3):432–456.

Omicini, A., Ricci, A., Viroli, M., Castelfranchi, C., and Tummolini, L. (2004). Coordination artifacts: Environment-based coordination for intelligent agents. In *Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'04)*, pages 286–293. IEEE Computer Society.

Parunak, H. V. D. and Weyns, D. (2007). Guest editors' introduction, special issue on environments for multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):1–4.

Sedaghat, M. N., Nejad, L. P., Iravanian, S., and Rafiee, E. (2006). Task allocation for the police force agents in robocuprescue simulation. In *RoboCup*, volume 4020 of *LNCS*, pages 656–664. Springer.

Yen, L.-H., Yeh, T.-T., and Chi, K.-H. (2009). Load balancing in IEEE 802.11 networks. *Internet Computing*, 13(1):56–64.