The benefits of multi-agent systems in spatial reasoning

Christian Kray
German Research Center for Artificial Intelligence (DFKI) GmbH, Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
kray@dfki.de

Abstract

This paper presents arguments for the application of a multi-agent approach to spatial reasoning, and it is shown how spatial reasoning benefits from an agent-based implementation. In order to achieve this, the basic processes in spatial reasoning are analyzed and mapped onto a multi-agent architecture. Additionally, common problems in the context of real-world applications are identified, and it is shown how these can be addressed using the proposed approach. Furthermore, it is argued that a realization based on a multi-agent system offers advantages in terms of extensibility and flexibility. Finally, an exemplary implementation of a spatial reasoning system within an interactive tourist guide is presented along with its basic mechanisms and a short review of its performance in real world tasks.

Introduction

Spatial cognition is a central component of human intelligence, and it is involved in many every day tasks such as wayfinding, locomotion, and vision. Spatial concepts are also very important for communication, e.g. when describing spatial constellations to a listener so that he can identify a certain object. Many approaches for enabling systems to reason about space have been proposed, ranging from strictly qualitative calculi (e.g. based on topological relations (Egenhofer & Herring 1990)) to more quantitative methods (e.g. using potential fields to compute linguistic descriptions of spatial constellations (Gapp 1994)).

There is a great number of possible applications (Cohn 1996) for these methods ranging from verbal interfaces for the blind to autonomous robot navigation. Additionally, the advent of affordable positioning systems (e. g. GPS, Gallileo) and the widespread adoption of mobile devices (such as mobile phones and PDAs) has sparked a great demand for location-aware services, and consequently for spatial reasoning. While this has led to the development of smart infrastructures for this kind of services ((Hohl *et al.* 1999), (Butz, Baus, & Krüger 2000)) - facilitating the handling of nomadic users - there are still some major hurdles that hinder the widespread application of spatial reasoning.

On one hand, the computations that have to be performed to evaluate spatial constellations and to arrive at some suitable solution are highly complex and time consuming. Moreover, the fundamental data upon which the reasoning takes place is frequently provided in a low-level metrical format (e. g. vector data in a GIS), while in many cases a symbolic or qualitative format is needed. On the other hand, the computational methods have to be adapted to take into account real-world characteristics (such as the current context). These points will be more thoroughly discussed in the following section along with a possible solution using multi-agent systems (MAS).

The remainder of this article is structured as follows. The next section shortly reviews the current links between spatial cognition and agent-based systems. After this, several reasons are highlighted for the adoption of a multi-agent approach in spatial reasoning. The following section describes an implementation of a spatial reasoning system along the lines of the points made in the previous section. The final section summarizes the ideas presented here, and provides some ideas for future research directions.

Basing spatial reasoning on multi-agent systems

In the context of spatial reasoning, the term *agent* is mostly used to describe a system that performs a specific task such as route advice (Rogers, Fiechter, & Langeley 1999) or incremental route instructions (Maaß, Baus, & Paul 1995). While these certainly meet the definition of an agent - which Wooldridge describes as *computer systems that are capable of autonomous action in some environment in order to meet their design objectives* (Wooldridge 1999) - they are not *multi*-agent systems per se. Multi-agent systems can be defined as systems that are *designed and implemented as several interacting agents* (Jennings, Sycara, & Wooldridge 1998). Although there are a few systems that take a decompositional approach to spatial cognition and exploit the benefits of parallel or resource-aware computations (e. g. (Blocher 1999)), the advantages of a true multi-agent system in spatial reasoning have not yet been thoroughly examined. In the following sections, several reasons are presented for the application of multi-agent systems generally, as well as arguments for modeling spatial reasoning using a multi-agent approach.

General qualities of multi-agent systems

Before looking at the specific advantages in the context of spatial-reasoning, some general qualities of multi-agent systems can be identified. (Bond & Gasser 1988) list several rationales for distributing an AI system, such as adaptability, cost-effectiveness, and improvements in the development and management process. In addition, they point out that the inherent isolation/autonomy of the parts provides not only protection for local information, but may also be a more 'natural' way to address certain problems. Furthermore, they argue that the distribution facilitates specialization, and may increase the reliability, robustness, and/or efficiency of the entire system. Finally, they state that resource limitations can be handled both on an individual, and on a larger scale. But before these benefits can be exploited in spatial reasoning, it is necessary to analyze how the reasoning process can be mapped onto agents.

Mapping processes onto agents

When looking at the different computational models for spatial reasoning ((Cohn 1996), (Mukerjee 1998)), one can divide the reasoning process into several distinct but interacting sub-processes. Many problems related to spatial constellations call for the evaluation of relations of some kind between some objects, or conversely, for the mapping of relations to a specific constellation. Depending on the set of relations in question, there may also be the need to establish a frame of reference (e.g. when considering angular relations). In the context of navigation, additional processes can be identified. In order to reason about routes, it often makes sense to divide them into segments, which can be mapped onto qualitative descriptions (Tversky & Lee 1999). In this case, a different body of relations - which relies to some degree on the shape of the objects involved ((Kray & Blocher 1999), (Mathet 2000)) - can also be taken into account. Figure 1 tries to summarize the precedent reflections by depicting these fundamental processes in the center part:

- **Two-Point Relations**
  This includes the evaluation of topological, angular and distance-dependent relations, which share the common characteristics that they - in their most basic form - relate two objects to each other[1].
- **N-Point Relations**
  Relations that cannot be reduced to a two-point-problem, such as path relations (Kray & Blocher 1999), special cases like *inbetween*, or constellational relations (Mathet 2000), are qualitatively different from two-point relations. They require more than two objects, and/or additional arguments such as shape or outline information.
- **Reference Object**
  Common tasks in the realm of spatial reasoning such as localization or object identification require the selection of a suitable reference object (or anchor object) from a set of possible candidates.
- **Frame of Reference**
  Several relations, especially angular relations depend on the establishment of a frame of reference in order to be non-ambiguous. Some calculi also use the frame of reference to scale space accordingly (Gapp 1995).
- **Route Segmentation**
  In the case of reasoning in the context of navigation or route descriptions, a complex route might have to be divided into several segments to allow for proper reasoning.

These processes do have to interact heavily in order to solve spatial problems. Consider, for example, the selection of a reference object. To identify the most appropriate one for the task at hand, static characteristics are often not sufficient. The relation chosen to go along with the reference object as well as whether it has been mentioned before contribute to its overall rating. The process of selecting a frame of reference also requires input from other processes, as the current reference object or the actual location of the user have a great influence on the outcome of the selection process. The segmentation of a complex route is another example of the intensive interaction between these core processes: a chosen subdivision depends not only on characteristics of the route , but is also determined by the quality of relations that can be established for the resulting parts, and by the context (e.g. at what speed a user is traveling).

Modeling these interacting processes as agents offers several advantages over a monolithic architecture. Firstly, the subprocess can be clearly separated, which allows for the concentration on the corresponding fundamental reasoning methods without having to take into account requirements of other subprocesses from the start. Secondly, the interaction between them is made explicit, which not only simplifies its formalization, but also makes it easy to change the interaction patterns (even dynamically) and facilitates the realization of meta-level reasoning. Thirdly, the agents are free to decide what they want to do next. This enables them, e. g. to reason about pending requests from other agents, and cancel obsolete jobs (such as the evaluation of a mediocre reference object if there is already a better one) or change the order of the processing of the jobs (e.g. if information required by a later job is already available). Finally, it is a straightforward task to extend and modify such a multi-agent system. Because the subprocesses are strictly separated, their internal reasoning mechanisms can be exchanged without disturbing the rest of the system. Adding new agents, that model different processes, can also be achieved rather easily, since the modifications required to let existing agents benefit from the new functionality are clearly encapsulated within them. In a monolithic architecture, isolating the places where changes have to be applied or changing the *implicit* interaction can be tedious and error prone.

Challenges posed by real-world applications

While these advantages apply to almost any spatial reasoning task, they do so especially in the context of real-world uses. When a user employs a system to solve a specific task, such as querying a geographical information system (GIS) to obtain some information, or asking for navigational

---

[1] Two points can be seen as the most basic entities, between which a relation can be established (Kray & Blocher 1999) - hence the name.

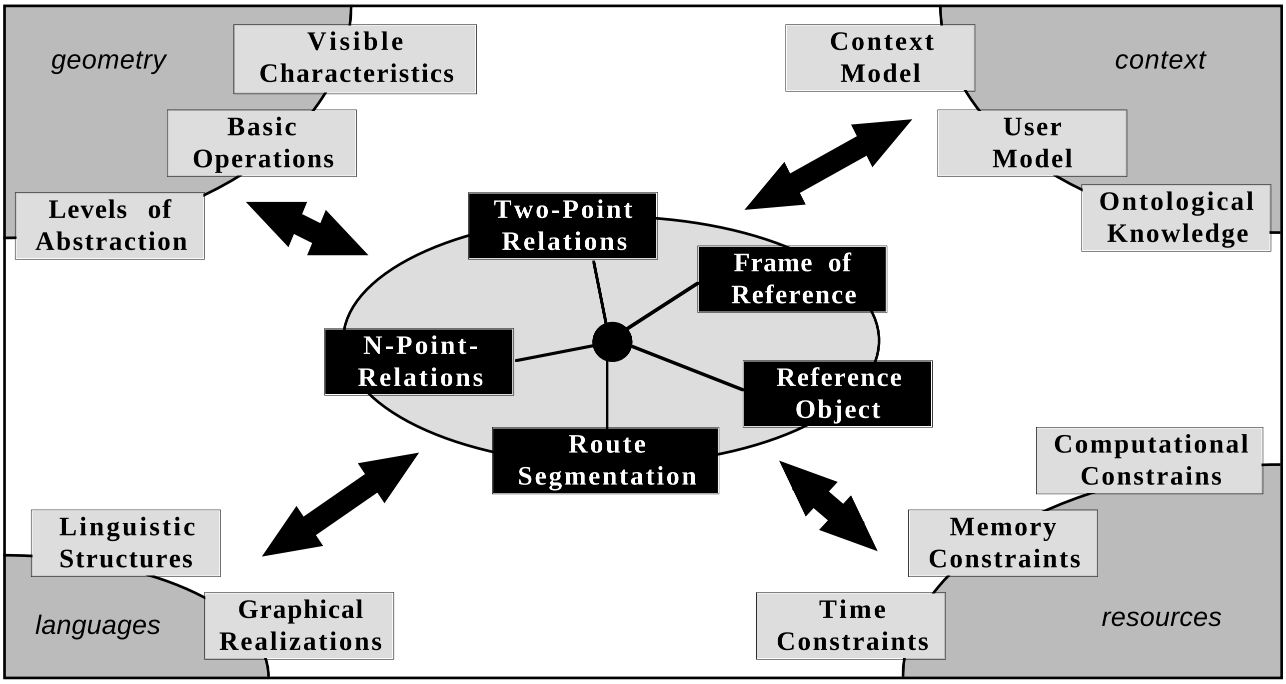

Figure 1: Processes, factors, and agents in spatial reasoning

instructions, the computational complexity of most spatial reasoning calculi (Cohn 1997) can become a big obstacle. Although there are methods, which can be solved in polynomial time, the great number of possible arguments (e.g. the number of objects within a GIS) and the amount of time to deduce qualitative information from the mostly metrical data (e.g. performing set operations on bitmap based data) require the application of heuristics, or at least some form of adaption strategy (such as anytime algorithms (Zilberstein 1993)), in order to keep answer times in a tolerable range.

A multi-agent approach helps to address these problems. On one hand, the freedom of decision that each agent has, allows for specific adaptation strategies and heuristics to be applied to the specific subproblems handled by the agent. One the other hand, it is an inherent ability of such systems to allow agents to duplicate themselves, or to move to a different site should they want to do so. This offers new ways to address the need for a timely solution: in case there is a large amount of information to be considered, several instances of a specific agent can concurrently work on small subsets. If the computational power of the current platform is insufficient to solve the task in time, an agent can decide to move to a more powerful site. Finally, the message passing mechanism, which is used for the inter-agent communication, makes it easy to realize a transactional paradigm: instead of waiting for the completion of a request sent to another agent, a specific agent may start working on a different task until it reaches a point where it again needs external information. After sending off the corresponding request, the agent can check whether there is already a reply to the first request. If that's the case, it can continue to work on the original task, thereby exploiting the 'waiting' time, which would have been wasted otherwise. These strategies can also be applied concurrently, to a single basic agent or to several ones, depending on the task at hand.

Furthermore, in a real-world context a great number of factors, that are usually left out of spatial reasoning, have to be included in the reasoning process. Figure 1 shows four fields, which play an important role in a real-world application: geometry, contextual factors, languages, and resource restrictions. While geometrical information in some form is present in almost any system concerned with spatial cognition, there are some factors that are especially important in a real world scenario, such as the visible characteristics of entities in the world or the level of abstraction, at which these entities are evaluated. These factors, or the reasoning/computation to extract them from the geometry of the world model, can again be encapsulated in agents, inheriting most of the benefits listed above and simplifying the interaction with the core processes described in the previous section. The same is true for contextual factors, which are particularly important in real-world applications. Knowledge about, e. g. which means of transportation is used (Maaß, Baus, & Paul 1995), what scale the user is referring to (Montello 1993), or what function a certain building has (Lynch 1960) strongly influence the outcome of the reasoning process. While the importance of resource considerations have already been discussed, the relevance of the languages which are used to inform the user of the results from the reasoning was not. But the fact, e. g. whether to realize the output graphically or verbally, and what specific (natural) language or symbols are selected can change the outcome of the reasoning[2].

This list of relevant factors for real-world spatial reasoning is certainly neither complete, nor has the *exact* way, in which they influence the reasoning process, been empirically examined. But building a system that includes these factors using a multi-agent architecture allows not only for a clear separation but also for continuous improvements: When new factors are identified, they can be added as an agent, and their influence on the other processes can be incorporated easily by first analyzing it individually for each process, and then establishing the interaction. Modifications to existing processes are on a local base without the need to search for all places, where changes would have to be made in a monolithic system. Furthermore, once empirical results become available about the way in which certain factors influence the reasoning processes, they can be implemented in a straightforward way. Either by simply changing the explicit interaction patterns, or by additional *local* modifications to the affected processes. In both cases, unaffected processes need not to be altered. These characteristics have been very beneficial in the development of the spatial reasoning system presented in the following section.

An exemplary implementation

The ideas described in the previous section have led to the realization of a spatial reasoning system SPACE[3], which implements many of the concepts presented here. It is part of a mobile tourist guide (DEEP MAP (Malaka & Zipf 2000)), where it is responsible for various tasks involving spatial problems, such as navigational instructions or queries with spatial constraints. Not only is SPACE part of a multi-agent system (the mobile tourist guide), but it also consists of agents itself[4]. The internal agents communicate with each other over an internal message bus, and can also send messages to external agents via the external message bus[5]. They model the basic processes identified in the previous section, and the spatial reasoning tasks, which arise in the context of the tourist guide (see figure 2).

These tasks include the identification (Identify) of objects in the environment of the user, and description (Describe) of objects the user mentions, e.g. when replying to queries like 'What's this?' or 'Tell me more about this building'. SPACE can also generate localizations for real-world entities using relational expression (Localize) and translate such expressions into geometrical regions (TransToGeo). Finally, there is an agent, that handles the incremental generation of navigational instructions (RouteManager), and a scheduling agent (Scheduler), which analyzes incoming jobs and distributes them among the internal agents.

During the two years of development, SPACE has been enhanced and extended frequently. As new information sources/agents became available, the internal agents were adjusted to include them in their reasoning. For example, when an agent for the management of the dialog history was introduced, the BestRO agent was modified to take into account whether an object had been previously mentioned when rating objects. The other agents did not have to be adjusted in order to benefit from this additional information. Adding an internal agent was also a straightforward task: When an agent for a hotel reservation system was introduced, the need arose for the translation of relational expressions (e. g. 'close to the station') to geometrical regions. TransToGeo was added, and did make use of the other existing agents without altering the system. The interaction between the agents is very intense, and would be very hard to handle if it were implicit. For example, when the BestRO agent evaluates an object, it contacts the GIS, the Dialog History, the Context Manager, and all other internal agents except Segmentation, and may engage in a message exchange with several of them. The same is true for the other agents of the system.

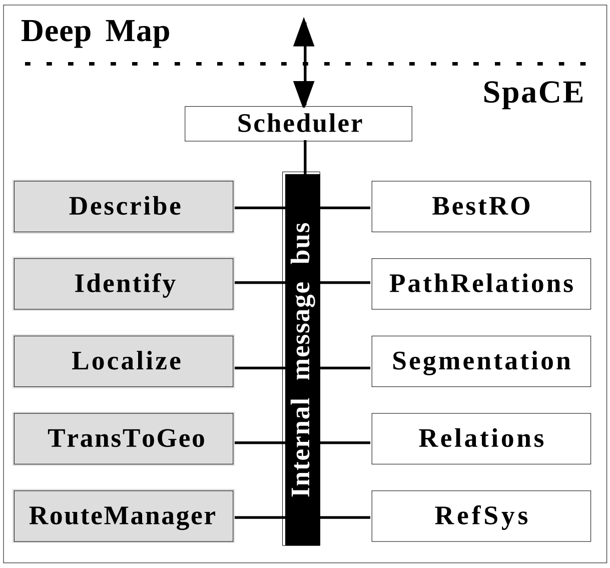

Figure 2: Architecture of SPACE

SPACE as it stands now can handle the tasks mentioned above within the city of Heidelberg (Germany) in a timely fashion. It relies on a GIS, which contains a detailed model of the entire city and its surroundings. The reasoning process takes into account information from a user model (e. g. age, knowledge about environment, physical constitution) and a context model (e. g. weather, means of transportation). The system also considers the user's current position (obtained from a GPS receiver) and facts from the dialog history. SPACE has been successfully used in a mobile prototype around the castle of Heidelberg. Currently, the system is being ported to a new agent platform, that offers enhanced resource monitoring and adaptation facilities, and a new version of the mobile prototype for systematic tests with casual users is prepared.

---

[2] For example, some relations might not be easily expressible verbally or graphically.

[3] **Spa**tial **C**ognition **E**ngine

[4] This architecture is often called *holonic* (Fischer 1999).

[5] This design ensures that internal message traffic is kept from the external bus, and external agents can address a single agent for all spatial problems.

Conclusion and Future Directions

This paper presented a new approach to the modeling of spatial reasoning processes using multi-agent systems. It was shown, that using the agent paradigm can be very beneficial in several aspects. On one hand, several basic processes can be identified and mapped directly onto agents, not only simplifying the model by encapsulating functionality, but also making the interaction between those processes explicit. On the other hand, the multi-agent approach facilitates the modification and extension of a spatial reasoning system, and is particularly well suited in the context of a real-world application since many additional factors have to be taken into account.

In the future, several research directions will be followed. Firstly, the interaction between the agents will be analyzed more thoroughly in order to formalize and optimize it. Secondly, a more rigid evaluation of the entire system is planned through systematic field tests with casual users. Thirdly, a component for user interaction about uncertain spatial knowledge (e.g. to determine the user's current location in case of sensor failure by means of a dialogue) is currently under development. Finally, additional resource adaptation strategies will be examined to further improve the average response time.

Acknowledgements

Deep Map, SpaCE, and the results presented in this paper are funded by the Klaus Tschira Foundation, Heidelberg. SpaCE is co-funded by the DFKI.

References

Blocher, A. 1999. *Ressourcenadaptierende Raumbeschreibung: Ein beschränkt-optimaler Lokalisationsagent (resource-adapting spatial descriptions: a bounded optimal localization agent)*. Ph.D. Dissertation, Computer Science Division, University of Saarland, Germany.

Bond, A. H., and Gasser, H. 1988. An analysis of problems and research in dai. In Bond, A. H., and Gasser, H., eds., *Readings in distributed artificial intelligence*. San Manteo, CA: Morgan Kaufmann.

Butz, A.; Baus, J.; and Krüger, A. 2000. Augmenting buildings with infrared information. In *Proceedings of the International Symposium on Augmented Reality ISAR 2000*. Los Alamitos, CA: IEEE Computer Society Press.

Cohn, A. G. 1996. Calculi for qualitative spatial reasoning. In Calmet, J.; Campbell, J. A.; and Pfalzgraf, J., eds., *Artifical Intelligence and Symbolic Mathematical Computation (LNCS 1138)*. Berlin: Springer. 124–143.

Cohn, A. G. 1997. Qualitative spatial representation and reasoning techniques. In Brewka, G.; Habel, C.; and Nebel, B., eds., *KI-97 (LNAI 1303)*. Berlin: Springer.

Egenhofer, M. J., and Herring, J. R. 1990. A mathematical framework for the definition of topological relations. In Brassel, K., and Kishimoto, H., eds., *Fourth International Symposium on Spatial Data Hadling*.

Fischer, K. 1999. Agent-Based Design of Holonic Manufacturing Systems. *Journal of Robotics and Autonomous Systems*.

Gapp, K.-P. 1994. Basic meanings of spatial relations: Computation and evaluation in 3d space. In *Proceedings of AAAI-94*.

Gapp, K.-P. 1995. Processing spatial relations in object localization tasks. In *AAAI Fall Symposium on "Computational Models for Integrating Language and Vision"*.

Hohl, F.; Kubach, U.; Leonhardi, A.; Rothermel, K.; and Schwehm, M. 1999. Next century challenges: Nexus - an open global infrastructure for spatial-aware applications. In *Proc. of Mobicom 99*. Seattle, Washington: ACM.

Jennings, N.; Sycara, K.; and Wooldridge, M. 1998. A roadmap of agent research and development. *Journal of Autonomous Agents and Multi-Agent Systems* 1:275–306.

Kray, C., and Blocher, A. 1999. Modeling the basic meanings of path relations. In *Proceedings of the 16th IJCAI. Morgan Kaufmann, San Francisco, CA*, 384–389.

Lynch, K. 1960. *The image of the city*. Cambridge, MA: MIT Press.

Maaß, W.; Baus, J.; and Paul, J. 1995. Visual grounding of route descriptions in dynamic environments. In *AAAI Fall Symposium on Computational Models for Integrating Language and Vision*. MIT, Cambridge, MA: AAAI.

Malaka, R., and Zipf, A. 2000. Deep Map - challenging IT research in the framework of a tourist information system. In Fesenmaier, D. R.; Klein, S.; and Buhalis, D., eds., *Information and communication technologies in tourism 2000*. Wien: Springer. 15–27.

Mathet, Y. 2000. New paradigms in space and motion: A model and an experiment. In Rodriguez, R. V., ed., *Workshop on Current Issues in Spatio-Temporal Reasoning at ECAI 2000*.

Montello, D. 1993. Scale and multiple psychologies of space. In Frank, A. U., and Campari, I., eds., *Spatial Information Theory: A Theoretical Basis for GIS*. Berlin: Springer-Verlag. 312–321.

Mukerjee, A. 1998. Neat vs scruffy: A survey of computational models for spatial expressions. In Olivier, P., and Gapp, K.-P., eds., *Computational Representation and Processing of Spatial Expressions*. Kluwer Academic Press.

Rogers, S.; Fiechter, C.-N.; and Langeley, P. 1999. An adaptive interactive agent for route advice. In *Proceedings of Autonomous Agents '99*, 198–205. New York, New York: ACM Press.

Tversky, B., and Lee, P. U. 1999. Pictorial and verbal tools for conveying routes. In *Spatial Information Theory (Proceedings of COSIT 99). Springer, Berlin*, 51–64.

Wooldridge, M. 1999. Intelligent agents. In Weiss, G., ed., *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*. The MIT Press. 21–35.

Zilberstein, S. 1993. *Operational Rationality through Compilation of Anytime Algorithms*. Ph.D. Dissertation, Computer Science Division, University of California at Berkeley.