AGENT-BASED COLLABORATIVE AND PARALLELED DISTRIBUTED GIS

LUO Yingwei *, WANG Xiaolin and XU Zhuoqun

Dept. of Computer Science and Technology, Peking University, Beijing, P.R.China, 100871 – lyw@pku.edu.cn

KEY WORDS: Geo-Agents, Collaboration, Parallel

ABSTRACT:

The collaboration and parallel mechanisms in a spatial multi-agent system Geo-Agents are explored and analysed in details. GIS agent in Geo-Agents are designed as four types: system manager – Facilitator, users or applications oriented agent - Interface Agent, spatial task undertaker – GIS Function Agent, and spatial database manager - GuServer. In Geo-Agents, Collaboration is embodied in two aspects: (1) collaboration among interface agents. Three types of collaboration among interface agents are discussed: direct cooperation, peer group cooperation and charge-tributary group cooperation. (2) collaboration under control of interface agent. Geo-Agents likes a kind of cluster parallel system, and parallel is performed by GIS function agents. Parallel of GIS function agents takes place in collaboration under control of interface agent, and exists as three modes: isomorphic cooperative parallel, heterogeneous cooperative parallel and exclusive parallel.

1. INTRODUCTION: GEO-AGENTS

Geo-Agents is an agent-based distributed GIS (see Figure 1). Geo-Agents consists of four types of GIS agent: Facilitator, Interface agent, GIS function agent and GuServer (Luo Yingwei, 1999; Luo Yingwei, *et al*, 2002).

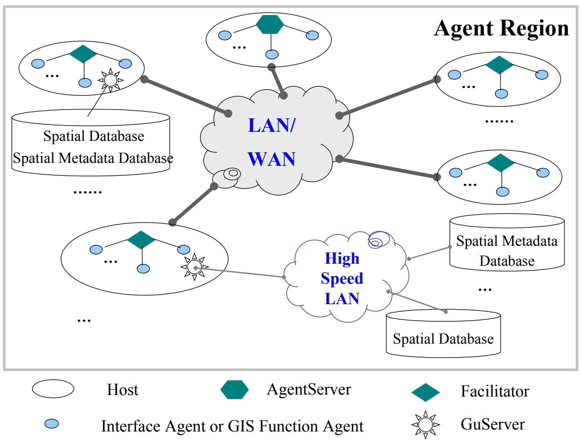

Figure 1 Geo-Agents System

Facilitator is the manager of Geo-Agents. The functions of Facilitator include registering available GIS agents, searching for practicable GIS agents, managing all active agent instances, coordinating communication and coordinating cooperation.

GIS function agent encapsulates spatial querying, spatial processing or spatial analyzing services. The encapsulated services in GIS function agent may come from different existing GIS platforms. Each GIS function agent can complete a same type of problem. According to the features of GIS, GIS function agent is classified into another two types: basic function agent and domain-oriented function agent. Basic function agent completes basic GIS services, such as spatial data search, spatial data access, network analysis, overlay analysis, buffer analysis and so on. Domain-oriented function agent is responsible for application tasks in various domains, and can be constructed by domain-oriented model and used generally in one domain.

Interface agent provides interfaces for users or applications to hand task. Geo-Agents provides GeoScript, an agent manipulating language to describe GIS tasks (Luo Yingwei, 1999). When solving a practicable problem, users or applications can simply use GeoScript statements to describe the task, and then hand the statements to interface agent. Interface agent has a GeoScript interpreter and can disassemble the task to subtasks autonomously, recruit GIS function agents to complete the task concurrently.

GuServer is in charge of spatial information accessing services, which manages spatial information and spatial metadata in spatial databases.

In Geo-Agents, GIS agent is reactive agent, and every GIS agent consists of five units: control subsystem, functional subsystem, communication subsystem, human-computer interface and data resource. GIS agent is not only able to carry out its own task independently, but also communicate with other agents, exchange information and cooperate with others.

"Agent Region" mode is adopted to control the distributed scenario for Geo-Agents. An "Agent Region" consists of one or more hosts, which must be installed with a Facilitator (and or other GIS agents). There are many Facilitators in an "Agent Region". Different Facilitators can cooperate to control and coordinate every GIS agent to run correctly, and hold the distributed controls of the whole system.

In an "Agent Region", there is one and only one Facilitator that will be configured as AgentServer. Facilitator is used to manage and coordinate agents inside one "Agent Region". Besides the functions of Facilitator, AgentServer serves as a bridge among different "Agent Regions". That is to say, an agent in one "Agent Region" can only communicate with agents in other "Agent Regions" through AgentServer. Of course, an

* Corresponding author: LUO Yingwei, lyw@pku.edu.cn.

authorization is required. AgentServer manages a table to register other "Agent Regions" that are authorized with itself each other. Between two directly authorized "Agent Regions", agents between them can communicate via the coordination of their AgentServers. The authorization relation in Geo-Agents can be passed one by on: If a directly authorized AgentServer chain can be found for two "Agent Regions" that have no direct authorization, agents between the two "Agent Regions" can communicate via the coordination of the AgentServer chain.

2. COLLABORATION MECHANISM IN GEO-AGENTS

In human society, there are some relations among individuals, which make individual collection become human society, and make individuals become socialization people. Just like human society, multi-agent system must let its agents cooperate with each other to become an integrated system, so as to complete more complicated tasks (Hyacinth S. Nwana, 1996; Ding Xiaoming and Liu Boqin, 1999).

The goal of Geo-Agents is to construct a multi-agent system like human society, so collaboration mechanism is necessary. There are two collaboration modes in Geo-Agents: collaboration among interface agents and collaboration under control of interface agent.

2.1 Collaboration among Interface Agents

Collaboration among interface agents is coordinated by AgentServer. AgentServer manages a table to register different cooperation groups in one "Agent Region". There are three types of collaboration among interface agents: direct cooperation, peer group cooperation and charge-tributary group cooperation.

(I) Direct cooperation: during the execution of an interface agent, it needs some results of other interface agents (see figure 2(a)).

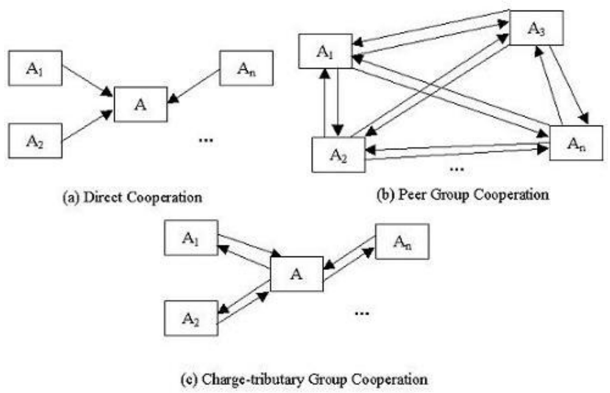

Figure 2 Collaboration among Interface Agents

(II) Peer group cooperation: several interface agents form a team. All team members cooperate to complete a same complex task. Each member assumes a subtask, and accomplishes it independently. Team members are peer in the team: after completing subtask, each member sends its result to all other members, so each member can obtain a same final result of the task (see figure 2(b)).

(III) Charge-tributary group cooperation: like peer group cooperation, all team members cooperate to complete a same complex task. But there is a team-charger. Team member completes subtask and sends its result to team-charger. Team-charger assembles all sub-results to final result of the task and sends the final result to all team members (see figure 2(c), where A is team-charger).

In Geo-Agents, the collaboration among interface agents still has some restrictions: (1) Collaboration exists inside only one "Agent Region". Collaboration among different "Agent Regions" is not supported now. (2) Each team member can communicate with others only once in one group. (3) The result type of each team member is same in one group, and all members cooperate to complete a same task. (4) In one group team, the number of team member is pre-fixed, and no team member can join into the team dynamically. (5) Communication among interface agents in one group can take place only after all team members are joined in.

2.2 Collaboration under Control of Interface Agent

After an interface agent accepts a complex task, it will recruit some GIS function agents, and then organize them to complete the task cooperatively. This is hierarchy collaboration under control of interface agent (see figure 3).

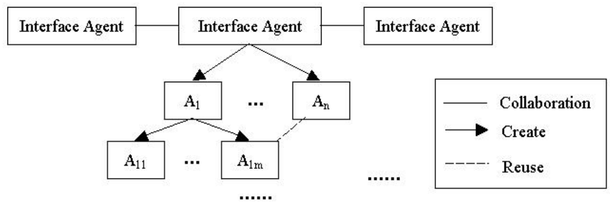

Figure 3 Collaboration under Control of Interface Agent

The hierarchy of agents is similar to human society. Different agents play different roles in the system when cooperating to complete a same task. Interface agent locates at top-level. The up-level agents may create some under-level agents or reuse other existing agents to complete subtasks. Those agents who belong to a same parent can execute sequentially or concurrently.

In hierarchy collaboration: 1) Facilitator/AgentServer is the collaboration coordinator; 2) up-level agents assign subtasks to down-level agents and down-level agents return to up-level agents; and 3) the results of some agents may be reused by others.

3. PARALLEL MECHANISM IN GEO-AGENTS

In a multi-agent system, the smallest executing unit is agent, and many agents can execute independently and concurrently. Geo-Agents likes a kind of cluster parallel system, and parallel is performed by GIS function agents. Parallel of GIS function agents takes place in collaboration under control of interface agent, and exists as three modes: isomorphic cooperative parallel, heterogeneous cooperative parallel and exclusive parallel.

(I) Isomorphic cooperative parallel: parallel agents are of the same type, and each of them completes a subtask of a same complex task. Assembling all subtask-results will get the final result. Isomorphic cooperative parallel meets two situations: one is that all agents process different data resource; the other is that all agents process different parts of a same data resource.

(II) Heterogeneous cooperative parallel: it is similar to isomorphic cooperative parallel, but parallel agents are of different types.

(III) Exclusive parallel: parallel agents are of same type, and complete a same task. But data resource each agent processes is different. Some agents may success, and some will fail. Of course, none knows aforehand which one will success or not. Only one successful result is needed. Once an agent returns successful result, all other agents must stop forcibly. If all agents fail, the task will fail too.

Although parallel is a intrinsic feature of multi-agent system, how to design parallel algorithms and construct suitable GIS function agents for real GIS problems in Geo-Agents still must be paid more attention.

Spatial data is often involved a large spatial scope, and the content is diversity. So people store spatial data in different places according spatial scope and content. Distribution is an intrinsic feature of spatial data. But in another hand, massive related spatial data for a domain is always stored in a same spatial database, so spatial data has another feature of centralization.

The distribution of spatial data brings many inconveniences because a spatial task always uses many kinds of spatial data from different spatial databases. But in another hand, just because of the distribution, many GIS function agents can cooperate to complete a same spatial task concurrently in different hosts. This strategy can make the best of distributed computing resources, and computing can be performed in the host where spatial data locates, so as to reduce the transferring quantity of spatial data in network.

Because of the centralization, the needed data resource for a problem may be in one host. A "Data Priority" strategy (an agent will try to execute in the host where data resource locates) is adopted in Geo-Agents, so although a GIS problems can parallel, the parallel agents will execute place in a same host. Because the computing resources are limited in one host and agents also exhaust some computing resources, this parallel will exhaust more computing resources than sequential execution by one agent. But from another viewpoint, although agent parallel in one host couldn't improve the performance, it provides a simplified and clear structure for constructing applications. For example, if there are two spatial metadata databases in one host, it is more convenient and clear to build two spatial metadata access agents to access different metadata databases respectively.

According to traditional viewpoint, parallel always improves performance. In fact, parallel is a kind of collaboration, but improving performance is not an intrinsic feature of collaboration. Firstly, collaboration brings new thought for software construction. Collaboration enables software construction organized as human society, so complex software construction can be built more easily and has a clearer architecture. Secondly, collaboration improves the capacity of software systems. For a software system, the capacity is primary. Collaboration makes some tasks achievable, which cannot be completed in traditional system. Only after a task can be complete correctly, improving performance is valid. Of course, aiming at the disadvantage brought by centralization of spatial data, a "peer hosts" mechanism is designed to partially improve the performance of Geo-Agents (Luo Yingwei, *et al*, 2002).

4. CONCLUSIONS

Collaboration and parallel processing on massive spatial information in network environment is a key problem that distributed GIS must face. Agent technology provides a new effective though and method for processing massive spatial information in network environment. Geo-Agents is an agent-based distributed GIS. The collaboration and parallel mechanisms in Geo-Agents are mainly designed according to the features of GIS and GIS applications. Through analyzing two collaboration and parallel samples in Geo-Agents, we can conclude that Geo-Agents can complete GIS tasks very well, improve the capability and performance of distributed GIS, and simplify the development of large complex GIS applications (Luo Yingwei, 1999).

There are too many collaboration and parallel factors in distributed GIS. Aiming at the real problems in GIS applications, how to design collaboration and parallel algorithms for massive spatial information processing and implement them in Geo-Agents will be emphases of our future work.

5. ACKNOWLEDGEMENT

Supported by the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312000; the National Research Foundation for the Doctoral Program of Higher Education of China under Grant No. 20020001015; the National Science Foundation of China under Grant No.60203002; the National High Technology Development 863 Program under Grant No. 2002AA135330 and No. 2002AA134030; the Beijing Science Foundation under Grant No.4012007.

6. REFERENCES

Luo Yingwei, 1999. A Study on Agent-based Distributed GIS. *PhD Dissertation* (in Chinese), Beijing: Peking University.

Luo Yingwei, *et al*, 2002. The Model of Distributed GIS-oriented Multi-agent System (in Chinese). *Acta Scientiarum Naturalium Universitatis Pekinensis*, 38 (3), pp.375-383.

Hyacinth S. Nwana, 1996. Software Agent: An Overview. *Knowledge Engineering Review*, 11(3), pp.205-244.

Ding Xiaoming and Liu Boqin, 1999. Cooperation Mechanism in MAS. *Computer Science* (in Chinese), 26(2), pp.54-56.