# REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188



| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | 13 February 2002 | Final Report, 10 August 2000 – 10 February 2002 |

4. TITLE AND SUBTITLE

Acquaintance Model Based Coalition Planning in Humanitarian Relief Operations

5. FUNDING NUMBERS

F61775-00-WE043

6. AUTHOR(S)

Michal Pechoucek, Professor Vladimir Marik, Jaroslav Barta

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Czech Technical University of Prague, CTU FEL, Technicka 2, Praha 6 166 27, Czech Republic

8. Performing Organization Report Number

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES

EOARD
PSC 802 Box 14
FPO 09499-0014

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

SPC 00-4043

11. SUPPLEMENTARY NOTES

Report carried out within the framework of the EPARD/US Air Force Research Contract “Acquaintance Models in Operations Other Than War Coalition Formation”, Contract No.: F61775-00-WE043, 38 Pages.

| 12a. DISTRIBUTION/AVAILABILITY STATEMENT | 12b. DISTRIBUTION CODE |
|---|---|
| Approved for public release; distribution is unlimited. | A |

ABSTRACT (Maximum 200 words)

This report results from a contract tasking Czech Technical University of Prague as follows: The contractor will investigate application of their collaboration acquaintance based multi-agent system (MAS) to Coalition Planning in Operations Other Than Was (OOTW). Application of MAS in coalition formation has already been studied. This effort will analyze the use of the Tri-Base Acquaintance 3bA) model as the formal model of agents’ mutual awareness in this domain. A research prototype (based on their experience implementing 3bA model in a MAS for project-driven production planning) will demonstrate and support the validity of their claims about the 3bA formal model of agents’ mutual awareness. This effort will (1) investigate potential roles of agent-based computing in OOTW, (2) investigate required functionality of possible multi-agent systems and (3) demonstrate the capacity of acquaintance models in practical, efficient and close-to-autonomous coalition formation and planning.

Abstract: The task of planning humanitarian relief operations within a high number of hardly collaborating and vague linked non-governmental organizations is a challenging problem. We suggest an alternative knowledge based approach to the coalition formation problem for humanitarian and peace-keeping missions. Owing to the very special nature of this domain, where the agents representing individual organizations may eventually agree to collaborate, but are very often reluctant to share their knowledge and resources, we tried to reduce the problem complexity by splitting the community of agents into alliances. We combined classical negotiation mechanisms with the acquaintance models and social knowledge techniques in order to reduce the communication traffic and to keep the privacy of knowledge. Experimental results are discussed in the paper.

| 14. SUBJECT TERMS | 15. NUMBER OF PAGES |
|---|---|
| EOARD, Humanitarian relief operation, Humanitarian mission, Peace-keeping mission, Organization collaboration, Communication traffic | |
| | 16. PRICE CODE |

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19, SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |

# Acquaintance Model Based Coalition Planning in Humanitarian Relief Operations

(Final report)

Michal Pěchouček, Vladimír Mařík, Jaroslav Bárta

Gerstner Laboratory, Department of Cybernetics,

Czech Technical University,

Technická 2, 166 27 Prague, Czech Republic

email: {pechouc|barta|marik}@labe.felk.cvut.cz

This document reports on the research that has been carried out within the framework of the EPARD/ US Air Force Research Contract "Acquaintance Models in Operations Other Than War Coalition Formation" (contract no.: F61775-00-WE043).

### Abstract

The task of planning humanitarian relief operations within a high number of hardly collaborating and vaguely linked non-governmental organizations is a challenging problem. We suggest an alternative knowledge based approach to the coalition formation problem for humanitarian and peace-keeping missions. Owing to the very special nature of this domain, where the agents representing individual organisations may eventually agree to collaborate, but are very often reluctant to share their knowledge and resources, we tried to reduce the problem complexity by splitting the community of agents into alliances. We combined classical negotiation mechanisms with the acquaintance models and social knowledge techniques in order to reduce the communication traffic and to keep the privacy of knowledge. Experimental results are discussed in the paper.

# Table of Contents:

# 1 Introduction

The application domain of this coalition formation research belongs to the area of **war avoidance operations** such as peace-keeping, peace-enforcing, non-combatant evacuation or disaster relief operations. Unlike in classical war operations, where the technology of decision making is strictly hierarchical, **operations other than war** (OOTW) are very likely to be based on cooperation of a number of different, quasi-volunteered, vaguely organized groups of people, non-governmental organizations (NGO's), institutions providing humanitarian aid, but also army troops and official governmental initiatives.

**Collaborative**, unlike hierarchical, approach to operation planning allows greater deal of flexibility and dynamics in grouping optimal parties playing an active role in the operation. New entities shall be free to join autonomously and involve themselves in planning with respect to their capabilities. Therefore any organization framework must be essentially "open". OOTW have, according to [25], multiple perspective on plan evaluation as there does not need to be one shared goal or a single metrics of the operation (such as political, economical, humanitarian). From the same reason, the goals of entities involved in a possible coalition may be in conflict. Even if the community members share the same goal, it can be easily misunderstood due to different cultural backgrounds.

The main reason why we can hardly plan operations involving different NGO's by a central authority results from their **reluctance to provide information** about their intentions, goals and resources. Consequently, besides difficulties related to planning and negotiation we have to face the problems how to assure sharing the detailed information. Many institutions will be ready to share resources and information within some well specified community, whereas they will refuse to register their full capabilities and plans with a central planning system and to follow centralized commands. They may agree to participate in executing a plan, in forming of which they played an active role. In our interpretation, an agent is a complex, organized entity (representing a NGO, humanitarian organization, army troop, etc.) playing an active role in the OOTW planning. A multi-agent system consists of a number of agents that group themselves in various, temporary coalitions (each solving a specific mission/part of the mission).

The main ambition of our research has been to analyze the problem of OOTW coalition formation and to propose a novel approach that would (i) make the coalition formation process simpler in comparison to the "classical" methods, and thus more efficient and (ii) at the same time maintain confidentiality of the private information. In our case, we decided to sacrifice the total optimality of the formed coalitions as we found this is not the most important aspect in the OOTW planning. We have suggested a concept of alliances – a set of agents that agreed to share some of their private information and to cooperate eventually. The coalition formation complexity is reduced by splitting the whole community of agents into disjunctive subsets (alliances) and by the attempts to create a coalition preferably within the single alliance. Social knowledge stored in the acquaintance models of individual agents has been widely explored in order to:

- **minimize required communication traffic** which influences the problem solving efficiency,
- **keep the quality of the coalition** that resulted from the coalition formation process operation 'reasonably good' – the quality has been measured by the humanitarian relief aid deliver time and by how much the coalition covers the request (in percent),
- **minimize the loss of agents' semiprivate information** when negotiating the mission – i.e. revealing the information about services the agent may provide, its status and intention in the minimum extent, and

– **minimize the amount of shared information** – information that possible coalition leaders know about other agents and use it in order to plan an optimal mission.

– **allow to reason about inaccessible agents** – analyze to which extend the social knowledge stored in acquaintance models may replace the inter-agent negotiation process.

The developed approach has been tested on the CPlanT multi-agent system implementation.

## 2 CPlanT System Architecture

CPlanT is a multi-agent system for planning humanitarian relief operations where any agent can initiate the planning process. Classical negotiation algorithms such as contract net protocol (CNP) [22] are used in combination with the acquaintance models techniques [6]. The CPlanT architecture consists of several specific classes of agents:

**Resource Agents** (R-agents) represent the in-place resources that are inevitable for delivering humanitarian aid, such as roads, airports. Unlike the below-defined H-agents, the R-agents are regarded as passive and they do not initiate any kind of humanitarian effort.

**In-need Agents** (In-agents) represent the centers of conflict that call for help (e.g. cities, villages, etc.).

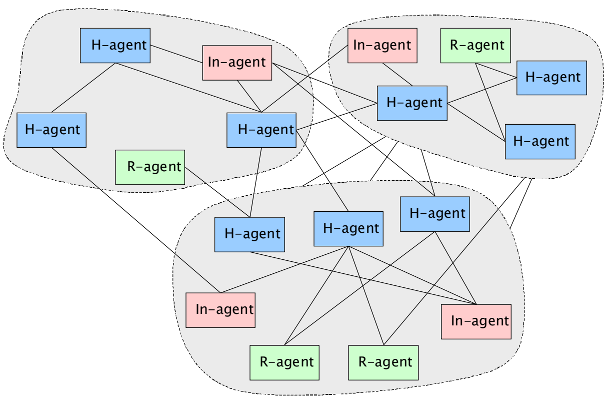


**Figure 1 – CPlanT Multi-Agent Architecture**

**Humanitarian Agents** (H-agents) represent the participating humanitarian agencies. Like the R-agents, the H-agents contribute to humanitarian aid missions. Therefore, one may regard the H-agent as a subclass of R-agents. However, the H-agents are proactive and they can initiate coalition formation process.

In this report, we will investigate coalition formation among the H-agents.

# 3 Knowledge Architecture

## 3.1 Agent's Neighborhood

Each H-agent may participate in one alliance of 'friendly' agents and at the same time it may be actively involved in several coalitions of agents cooperating in fulfilling specific shared tasks. Computational and communication complexity of forming such a coalition depends on the amount of pre-prepared information the agents administer one about the other and on sophistication of the agents' capability to reason about the other agents' resources, plans and intentions. The agents can allow others to reason about them and at the same time they can reason differently about the agents that belong to their different scopes of reasoning – neighborhood. Therefore, we distinguish among several types of agents' neighborhoods:

- $\alpha(A)$ – agent's **total neighborhood**, a set of all agents that the agent $A$ is aware of, (e.g. knows about their existence and is able to communicate with them)
- $\mu(A)$ - agent's **social (monitoring) neighborhood** that is a set of agents, which the agent $A$ keeps specific information about (e.g. services they provide, status, load, etc.). This neighborhood consists of the set of the agents about who the agent $A$ reasons and keeps knowledge about services they provide (status, load, etc.). According to [12] the agents social neighborhood consist of agents that the agent $A$ reasons about – $\mu^+(A)$ and the set the agents that reason about the agent $A$ – $\mu^-(A)$. Therefore

$$\forall\, B \in \mu^-(A): A \in \mu^+(B).$$

- $\varepsilon(A)$ – agent's **cooperation neighborhood** that is a set of agents jointly collaborating (or committed to collaboration) in achieving one or more shared goals.

## 3.2 Knowledge Sharing

In order to reason one about the other, the agents must share some of their knowledge. Let us introduce the operator ($\mathbf{Bel}\ A\ \varphi$) that expresses the agent's $A$ awareness of the formula $\varphi$ being true (Wooldirdge 2000). We say that the agent $A_0$intentionally shares its knowledge $\mathbf{K}(A_0)$ with a set of agents $\delta(A_0) \subseteq \Theta$ provided that:

$$\mathbf{K}(A_0) = \{\varphi\} : \forall\varphi \in \mathbf{K}(A_0) : \forall A_i \in \delta(A_0) : (\mathbf{Bel}\ A_i\ \varphi) \wedge$$
$$\forall B_i \notin \{\delta(A_0) \cup \{A_0\}\} : (\mathbf{Bel}\ A_0 \neg(\mathbf{Bel}\ B_i\ \varphi)).$$

From the previous follows, that if an agent $B$ knows some of the shared information without the agent $A_0$ being aware of this fact, the agent $B$ is not regarded as a member of the $\delta(A_0)$ set of agents, representing $A_0$'s knowledge sharing neighborhood. According to this classification, we suggest three levels of the H-agent's knowledge sharing:

**Public Knowledge** is shared within the entire multi-agent community. If it is assumed that all the agents know one about the other (i.e. $\forall A, A \in \Theta : \alpha(A) = \Theta$), public knowledge $\mathbf{K_P}(A_0)$ of an agent $A_0$ is defined as

$$\mathbf{K_P}(A_0) = \mathbf{K}(A_0) \text{ where } \delta(A_0)=\alpha(A_0).$$

This class of knowledge is freely accessible within the community. As public knowledge we understand the agent's name, the type of the organization the agent represents, the general objectives of the agent's activity, the country where the agent is registered, the agent's human-human contact (telephone, fax number, email), the human-agent type of contact (http address), the agent-agent type of contact (the IP address, incoming port, ACL) and, finally, available services.

**Semi-Private Knowledge** (also referred to as alliance accessible knowledge) is shared within agents' social neighborhoods. Semi-private knowledge $\mathsf{K_s}(A_0)$ of an agent $A_0$ is defined as

$$\mathsf{K_s}(A_0) = \mathsf{K}(A_0) \text{ where } \delta(A_0) = \mu(A_0).$$

As in the OOTW domain, we do not assume the knowledge to be shared within the overlapping alliances, we will require the social neighborhood to have the following property: $\forall\, A \in \Theta : \mu^{-}(A) = \mu^{+}(A) = \mu(A)$. Members of a social neighborhood share information about availability of their resources.

**Private Knowledge** is owned and administered by the agent itself. Private knowledge $\mathsf{K_P}(A_0)$ of an agent $A_0$ is defined as

$$\mathsf{K}_{pr}(A_0) = \mathsf{K}(A_0) \text{ where } \delta(A_0) = \{\}.$$

An important type of private knowledge includes agent's collaboration preferences, alliance restrictions, coalition leader restrictions and possible next restrictions, but also agent's planning and scheduling algorithms.

### 3.3 Alliance, Coalition, Team Action Plan

In the subject domain, we will understand as the multi-agent community $\Theta$ the whole collection of agents participating in the above-described OOTW task (quasi-volunteered, vaguely organized groups of people, non-governmental organizations, institutions providing humanitarian aid, army troops or official governmental initiatives). We will introduce the concept of an **alliance** as a collection of agents that share information about their resources and all agree to form possible coalitions. The alliance is regarded as a long-term cooperation agreement among the agents. Members of an alliance will all belong to one others' social neighborhood. Provided that we assume that each agent belongs also to its own social neighborhood – $\forall\, A \in \Theta: A \in \mu(A)$, we define the alliance as follows:

| |
|---|
| An **alliance** is a set of agents $\kappa$, so that $\forall\, A \in \Theta : \exists\kappa : A \in \kappa \wedge \forall\, A_i \in \kappa : \kappa = \mu(A_i)$. |

A singleton agent is regarded as an alliance with just one member. From the requirements for the reciprocal knowledge sharing within an alliance follows that

$$\forall\, A \in \kappa : \kappa = \mu(A).$$

Therefore, an important property of an alliance is that it cannot overlap with another alliances:

$$\forall\, \kappa_1, \kappa_2 \subseteq \Theta: (\exists A: A\in \kappa_1 \wedge A \in \kappa_2) \Rightarrow \kappa_1 \equiv \kappa_2.$$

Let us define a **coalition** as a set of agents, which agreed to fulfill a single, well-specified goal. Coalition members committed themselves to collaborate on the within-coalition-shared goal. Under the assumption $\forall A \in \Theta: A \in \varepsilon(A)$ we define a coalition as follows:

| |
|---|
| A **coalition** is a set of agents $\chi$, so that $\forall \chi(\tau) \subseteq \Theta: \forall\, A \in \chi(\tau) : \chi(\tau) \subseteq \varepsilon(A)$. |

Let us introduce a set $\varepsilon(A,\tau)$ that is an agent collaboration neighborhood with respect to a single shared goal $\tau$. Then

$$\varepsilon(A) = \bigcup_{\tau} \varepsilon(A,\tau), \text{ and}$$

$$\forall \chi(\tau) \subseteq \Theta: \forall A \in \chi(\tau) : \chi(\tau) = \varepsilon(A,\tau).$$

A coalition, unlike an alliance, is usually regarded as a short-term agreement between collaborative agents. As we will see in Section 6, it is better for a coalition to be a subset of one alliance, but it is not an inevitable condition. A coalition can consist of agents who are members of different alliances.

Another term that we have to introduce is a **team action plan**. In planning humanitarian relief operations, similarly as in the case of any other collaborative action planning, the agents must agree on how they will achieve the goal $\tau$. The team action plan is thus a decomposition of a goal $\tau$ into a set of tasks $\{\tau_i\}$. The tasks will be delegated within the coalition members. Apart from the responsible agent, each task shall be denoted by its due time, start time and price. Provided that an agent $A_j$ is responsible for implementing the task $\tau_i$(where $\tau = \{\tau_i\}$) in the time $\mathrm{due}(\tau_i)$, starting at $\mathrm{start}(\tau_i)$ for the price $\mathrm{price}(\tau_i)$, we define the team action plan as follows:

| |
|---|
| A **team action plan** $\pi(\tau)$ is as a set $\pi(\tau) = \{\langle \tau_i, A_j, \mathrm{start}(\tau_i), \mathrm{due}(\tau_i), \mathrm{price}(\tau_i)\rangle\}$. |

We say that the team action plan $\pi(\tau)$ is **correct** if all the collaborators $A_j$ are able to implement the task $\tau_i$ in the given time and for the given price. The team action plan $\pi(\tau)$ is **accepted** if all agents $A_j$ get committed to implementing the task $\tau_i$ in the given time and for the given price. Similarly, we say about the goal $\tau$ to be **achievable,** if there exists such $\pi(\tau)$ that is correct. The goal $\tau$ is said to be **planned,** if there exists $\pi(\tau)$ that is accepted. Obviously, there is an important relation between the team action plan and the coalition. We say that a coalition $\chi(\tau)$ achieves a goal $\tau$ by implementing a team action plan $\pi(\tau)$ if and only if $\chi(\tau)= \{A_j\}$ and $\pi(\tau)$ is correct.

### 3.4 Disclosure of Private and Semi-Private Knowledge

Measuring the loss of information, that the agents may want to keep private, is an uneasy task. The revealed piece of information has got different value to the agents with different meta-reasoning capabilities [12]. In order to vaguely categorize various types of information leaks, let us distinguish between strong and weak leaks.

- **Strong information disclosure**: If an agent looses some type of private (or semi-private) knowledge in the strong sense, it does so as a side effect of some proactive step (such as sending a request).
- **Weak information disclosure**: If an agent looses the private knowledge in the weak sense, it deliberately discloses some piece of its knowledge to other agents being asked for this specific piece (e.g. when sending an `inform`-type message).

Each agent undertakes the weak loss of some of its knowledge when forming an alliance. At this moment the agent's semi-private knowledge gets disclosed within the alliance members. The agent looses some of its private knowledge in the strong sense, if it communicates with an agent, which is outside of its alliance. Once the agent $A_1$ from an alliance $\kappa_1$ sends a request for a service $\tau$ to the agent $A_2$ from the alliance $\kappa_2$, the agent $A_1$ reveals the information about the **intent** (e.g. $A_1$ does something that requires $\tau$) and information about agent's $A_1$ capabilities (e.g. $A_1$ cannot do $\tau$). At the same time, a proposal for collaboration from the agent $A_2$ reveals the information about the agent's $A_2$ capabilities (such as $A_2$ can implement $\tau$ in time $t_1$). However, this type of knowledge disclosure has been reduced as the agent $A_2$ acts on behalf of the entire alliance. Therefore, if $A_2$ offers some services that are not used at the end, there is a loss of information about capabilities of the entire alliance (and not of the individual agent $A_2$ itself).

## 4 Agents' Acquaintance Model

Let us very briefly introduce the concept of agent's social intelligence and acquaintance models. Apart from its **problem-solving knowledge** that guides the agent's autonomous local decision making processes (such as coalition formation, or team action planning), the agents usually exploit **social knowledge** that expresses the other agent's behavioral patterns, their capabilities, load, experiences, resources, commitments, knowledge describing conversations or negotiation scenarios [11]. This knowledge is usually stored separately from the agents' computational core – in an agent's **acquaintance model**. There have been investigated several acquaintance models previously. Based on the *tri-base acquaintance model* [13], the social knowledge in CPlanT is organized in four separate knowledge structures:

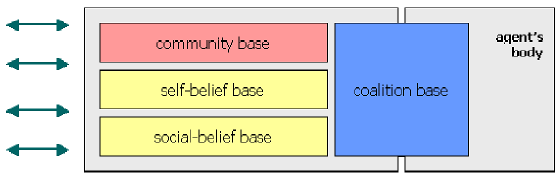


**Figure [illegible] – Structure of the CPlanT Acquaintance Model**

– **community-base** (Com–BB) – which is a collection of the community members' public knowledge

$$\text{Com–BB}(A_0)=\{K_P(A_i)\} \text{ for } \forall A_i \in \alpha(A_0)$$

– **self-belief-base** (Self–BB) – where the agent's reflective knowledge about itself is located; here the agent stores its public knowledge that is accessible to anyone, its semi-private knowledge that is shared within the alliance and its private knowledge that is not shared by anyone,

$$\text{Self–BB}(A_0)= \{\{K_P(A_0)\}, \{K_S(A_0)\}, \{K_{Pr}(A_0)\}\}$$

– **social-belief-base** (Soc–BB) – where the agent stores the semi-private knowledge of its peer alliance members,

$$\text{Soc–BB}(A_0)=\{K_S(A_i)\} \text{ for } \forall A_i \in \mu(A_0)$$

– **coalition-base** (Coal-BB) – which is a dynamic collection of the peer coalition members, the past and possible future coalitions as much as permanent coalition-formation rules[1].

Exploitation of the acquaintance model reduces communication traffic required for collaborative activity planning. In principle, the social knowledge substantially reduces the set of agents (ideally to one) that will be requested by the coordinating agent in the CNP process [22]. An important flaw of this approach is rooted in high requirements for the social model maintenance. The social knowledge maintenance may be driven either by the owner of the acquaintance model (the coordinator) or by those which are represented in the model – hence service providers (collaborators). We refer to the former strategy as the **requestor-driven** knowledge maintenance and to the latter strategy as the **provider-driven** knowledge maintenance. As an example of a **requestor-driven** strategy there is the concept of **periodical revisions** [10] where the knowledge owner periodically checks consistency of the model with the potential collaborators. In other systems, there has been a **cooperation trader** type of agent, which was in charge of maintaining the agent's social knowledge. We have adopted the **provider-driven** knowledge maintenance in CPlanT.

| | | |
|---|---|---|
| **Self-belief base** | | |
| **Public knowledge:** | Semi-private knowledge: | Private knowledge |
| **Port:** 1500<br>**ip address:** "147.32.86.167"<br>**Country:** suffer terra<br>**City:** north port<br>**Type:** Religious<br>**Ontologies:** fipa-am, cplant-ontology | Food: 3000<br>Nurses: 50<br>Trucks: 13 | Alliance restrictions: ("country","Suffer Terra")<br>Leader restrictions: ("type","Military").<br>City restrictions: ("muslim",50)<br>Cooperates with: ("type","government") |
| **Social belief base** | | |
| | **Agent:** ST Police | Armed-people:30 |
| | **Agent**: Christian STHO | Food: 3500 Clothing: 280<br>Nurses: 22 Medical-people: 12 |
| **Community belief base** | | |
| | **Agent:** Suffer Terra Government | Suffer Terra Government@iiop://147.32.84.131:2188/Suffer Terra Government<br>Type: Government<br>Services: Food, Civil-material, Medical-material, Clothing<br>Ontologies: FIPA-AGENT-MANAGEMENT, MAP-ONTOLOGY, PORT-ONTOLOGY, CPLANT, ALLIANCE<br>Languages: SL1, KIF, State: ACTIVE<br>Country: Suffer Terra, City: Suffer Town |
| | **Agent:** Christian STHO | Christian Suffer Terra Humanitarian Organization@iiop://147.32.84.131:2210/Chr ST Humanitarian Organization<br>Type: Religious<br>Services: Food, Clothing, Medical-people, Nurses, Medical-material<br>Ontologies: FIPA-AGENT-MANAGEMENT, MAP-ONTOLOGY, PORT-ONTOLOGY, CPLANT, ALLIANCE<br>Languages: SL1, KIF, State: ACTIVE<br>Country: Suffer Terra, City: North Port |
| **Coalition base** | | |
| **Rules:** | (VOLCANIC-AVERAGE-SMALL-TOWN → Time: 220 (Requirements: Medical-material 60, Food 1500, Civil-material 30000, Medical-people 16, Civil-people 27, Nurses 19) … | |
| **Coalitions:** | (coalition (**Members:** Suffer Terra Government, Suffer Terra Police, Christian Suffer Terra Humanitarian Organization)<br>(**Services:** Food, Civil-material, Medical-material, Clothing, Military-people, Food, Clothing, Medical-people, Nurses)<br>(**Price-for-coordination**: 5))<br>(planned-coalition ( Task name: Suffer-Town-24-1-2002/17-49-53.1<br>(Coalition members: Suffer Terra Government, Suffer Terra Police, Christian Suffer Terra Humanitarian Organization)<br>(Coalition leader: Christian Suffer Terra Humanitarian Organization<br>(Disaster: Volcanic, Degree: 1, (Allocations: Civil-material, 80000, Allocation Time: 350<br>Food, 80000, Allocation Time: 350<br>Medical-material, 80000, Allocation Time: 350)) … | |

**Table 1 – Instance of an [illegible]-agent's acquaintance model**

[1] The coalition-formation rules are instances of the agent's problem-solving knowledge, while the information about the coalition members, past and future coalitions are instances of the social knowledge. Therefore the coalition base belongs in part to both the acquaintance model and the agent's body

# 5 Inter-Agent Communication

Before explaining the lifecycle of the system, let us comment the main communication techniques that have been used in the CPlanT: the central communication agent, the contract net protocol, and the acquaintance models.

## 5.1 Central Communication Component

We have tried to minimize the use of a central communication component, as it is an important communication bottleneck of the system operation and the center where the agent's private knowledge may be sniffed and aggregated. For utilization of this communication technique see Section 6.1.

## 5.2 Contract Net Protocol

The CPlanT implementation relied heavily on the **contract net protocol** (CNP) negotiation scenario [22]. Any agent can initiate the coalition forming process (hereafter we refer to this agent as a coalition **coordinator**) by requesting some agents in the community (**collaborators**) for specific services. Upon receiving proposals for collaboration, the coordinator carries out a computational process by which it selects the best possible collaborator(s) – see Figure 3. The coalition planning process can also be multi-staged. Such an approach requires substantial computational resources and fails in complex communities. For each single-staged CNP within a community of $n$ agents, it is needed to send $2(n+1)$ messages in the worst case.

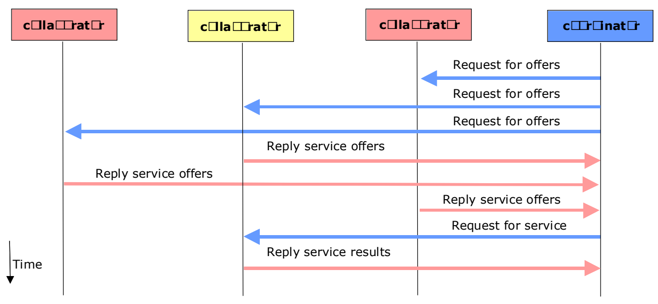


**Figure □– C□ntracti□n □a□e□□n a □ingle-□tage□C□ntract □et Pr□t□c□l**

At the same time many agents may not want to enter the CNP negotiation, as they wouldn't wish to undertake the risk of disclosing their private knowledge.

## 5.3 Acquaintance Model Contraction

An alternative communication strategy to CNP is based on exploitation of the agents' social knowledge. A coalition coordinator subscribes (by sending messages of the `subscribe`-type) the potential collaborators for specific services they may want to exploit in the future. Upon a change in the collaborators' capabilities, they provide the coordinator with an update in the form of a message of an `inform`-type. When the coordinator triggers the coalition formation phase, it parses the prepared service offers and selects the best collaborator(s) without any further negotiation. The coordinator sends a request, the collaborator updates its resources and confirms the contract. Any change in collaborator resources is advertised to all the coordinators which subscribed the collaborator (see Figure 4).

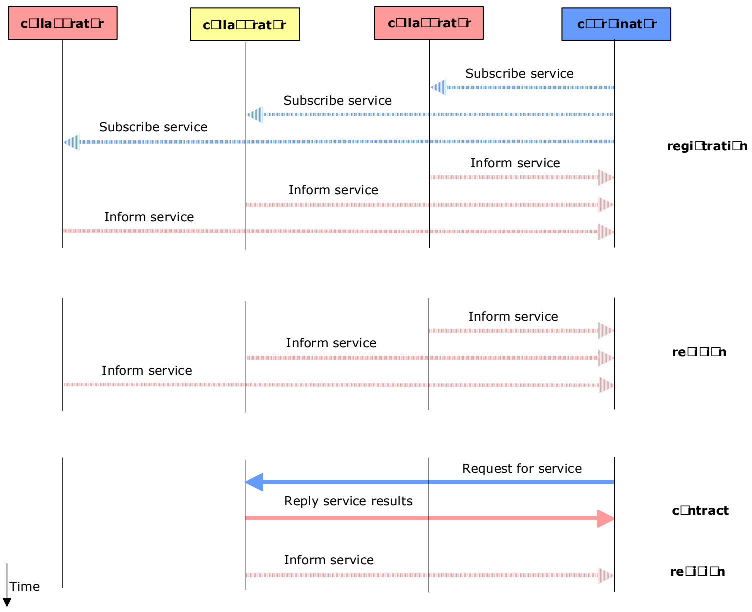


**Figure □– C□ntracti□n □a□e□ □n Ac□uaintance M□□el e□□□itati□n**

If there is a single event in the community Θ that affects all the agents ($n = |\Theta|$) and all the agents are mutually subscribed, then in the worst case there is ($n(n-1)$) messages required for the social knowledge maintenance on this event. However, this is rarely the case. Agents never subscribe all each other (we could easily use a central communication component instead).

## 6 CPlanT Operation Lifecycle

The CPlanT multi-agent system operates in four separate phases:

(i) **registration** for the agents' login/logout to/from the community,

(ii) **alliance formation** for forming of alliances,

(iii) **coalition formation** for finding a group of agents which can fulfill the specified task, and

(iv) **team action planning** for resource allocation within the specific coalition.

In the following, we will comment each of these phases in more detail.

As pointed above, we did not adopt one communication technique and claim that this is the best for our problem. We were trying to find a middle ground among the central communication agent technique, the contract net protocol scenario, and the acquaintance model based approach in order to optimize:

– communication traffic (and computational resources) requirements in both
  - the coalition formation and team-action planning phases, and
  - the periodic communication traffic in the agents' idle times (mainly ensuring the maintenance of the social models),

– quality of the formed alliance, the coalition and primarily the team action plan, and
– the amount of the private information that the agents have to disclose when forming the coalition.

### 6.1 Registration

Throughout the registration phase, a new-coming agent registers within the multi-agent community. This agent registers its public knowledge with the special central registration agent – the **facilitator.** Subsequently, the facilitator informs all the already existing agents about the new agent, and it also informs the new agent about all existing agents. After the registration phase, all the agents will be aware of the other existing agents, formally:

$$\forall A, A \in \Theta: \alpha(A) = \Theta.$$

Similarly, the agents can deregister with facilitator. Any registered agent stores the public knowledge about all the members of its total neighborhood $\alpha(A)$ that has been stored in the Com-BB($A$) bases of agents' acquaintance models.

**Communication strategy**: Any multi-agent system cannot go by without at least a tiny bit of centrality. We have used the **central communication unit** – directory facilitator in the registration phase. As the agents register only their public knowledge, we do not breach the requirements for confidentiality of the private information.

### 6.2 Alliance Formation

In this phase, which follows the registration process, the agents analyze the information they have about the members of the multi-agent system and attempt to form alliances. In principle, each agent is expected to compare its own private knowledge (i.e. alliance formation restrictions) with the public knowledge about the possible alliance members (i.e. type of an organization, its objectives, country of origin, etc.). Had the agent detected a possible future collaborator, the agent would propose joining the alliance. Throughout the negotiation process, the agent either chooses the best alliance according its collaboration preferences of agents into already existing alliances. Failing to do so, an agent may start a new alliance by itself.

According to their preferences in Self-BB and the community public knowledge in Com-BB, the agents carry out a selective contract net protocol process during this phase. The **quality of an alliance** is understood in terms of maximizing the individual agent's contribution to the alliance (i.e. covering the biggest amount of services that the other members of the alliance cannot implement). It is important to note that this process does not give us any guarantee for optimality of the alliance allocation. Each agent will join the most profitable alliance with respect to the existing alliance configuration. With changing the order of agents' registration with the alliance, the formation algorithm will come up with different alliances.

```
join-alliance(A)
  μ'(A) ← find-collaborators(Self-BB(A), Com-BB(A))
    ∀ Ai ∈ μ'(A)
    send-message(perform: request, sender: A receiver: Ai, content: join-alliance-with(A))
   μ''(A) ← {Ai}, where
        receive-message(perform: agree, receiver: A, sender: Ai, content: μ(Ai)) ∧ ∀ Bi ∈ μ(Ai): Bi ∈ μ'(A)
   μ(A) ← choose-the-best( ⋃ μ(Ai) ∪ {Ai}),
          where ∀ Bj ∈ μ(Ai) ∪ {Ai}: Bj ∈ μ''(A)
   ∀ Ai ∈ μ(A)
    send-message(perform: confirm, sender: A receiver: Ai, content: join-alliance-with(A))

 }

on
receive-message(perform: request, sender: Ai receiver: A, content: join-alliance-with(A))
 { if A ∉ alliance-membership-restrictions(Ai)
       then return 'AGREE ∧ μ(Ai)
       else 'REFUSE
 }
```

**Figure 5 – Algorithm of the alliance formation process in a hypothetical language**

See Figure 5 for an algorithm of the alliance formation process. The upper part describes behavior of the agent once it is registered. After having parsed its Self-BB and Com-BB of its acquaintance model, the agent broadcasts proposals for joining alliances and selects/joins the most suitable alliance. The lower part illustrates behavior of the agent who was asked to accept a new alliance member. It deliberates using simple rules – alliance-membership-restrictions.

### 6.3 Coalition Formation

In this phase, the agents do not group together according to similar general mission objectives, but they form coalitions with respect to a single, well-specified particular task that needs to be accomplished. Both, the CNP technique and the acquaintance model have been used in the coalition formation process. First, let us talk about the coalition formation process within a single alliance. The alliance members know the most of each other and are able to suggest a coalition that will very likely have the foreseen properties. Whichever agent, member of an alliance, can play the role of the coordinator of the goal $\tau$ implementation. The coordinator, who is to be set randomly in our implementation, parses its social neighborhood $\mu(A)$ and detects the set of the most suitable collaborators (cooperation neighborhood) – $\varepsilon(A, \tau)$. Upon an approval from each of the suggested agents, the respective coalition $\chi(\tau) = \varepsilon(A, \tau)$ is to be formed. Maintaining the agents' social neighborhoods will save an important part of the agent's interaction in the time of coalition formation. Agents will not need to broadcast a call for collaboration each time they will be required to accomplish a task. Instead, they will consult this pre-prepared knowledge and will contract the agent about which they knew it is the best to work with. The coordinator optimizes the **quality of a coalition** by seeking the coalitions that would contribute the most and in the shortest possible time.

As said in the previous, the agents' prefer not to form coalitions across alliances ($\forall \tau: \varepsilon(A, \tau) \subseteq \mu(A)$). However sometimes an alliance fails to achieve a goal. In such a case, the goal $\tau$ is achievable so that

$$\forall \pi(\tau): \neg\exists\kappa \supseteq \{A_j\}, \text{ where } \tau = \{\langle \tau_i, A_j, \mathrm{start}(\tau_i), \mathrm{due}(\tau_i), \mathrm{price}(\tau_i)\rangle\}.$$

The coordinator who failed to form a coalition within one alliance uses the classical CNP negotiations and broadcasts a proposal for collaboration to the agents from its total neighborhood $\alpha(A_0)$[2].

```
form-coalition(A, τ)
        {χ1(τ) := find-non-possible-coalitions(τ, μ(A), social knowledge)
         form-coalition-r(A, τ, χ1(τ))
        }
form-coalition-r(A, τ, χ1(τ))
        {repeat {χ(τ) := find-the-best-coalition(τ, μ(A), social knowledge )}
         until χ(τ)∉χ1(τ)
         if χ(τ) == {} then return failure
         else ∀Ai ∈ χ(τ)
      send-message(perform: request, sender: CL receiver: Ai, content: χ(τ))
           if ∀Ai ∈ χ(τ)
      recieve-message(perform: reply, sender: Ai, receiver: CL, content: "AGREE")
             then return(coalition-leader: A, coalition χ(τ))
             else
      form-coalition-r(A, τ, reduce-coalition(χ(τ)) ∪ χ1(τ))
}

on receive-message(perform: request, sender: A receiver: Ai, content: χ(τ))
 {if χ(τ) ∈ acceptable
        then return "AGREE"
        else "REFUSE"
 }
```

**Figure 6 – Algorithms for the coalition formation process**

See Figure 6 for an algorithm of the deliberation process when forming a coalition by the coalition lead. The state-space is reduced by finding the illegal coalition specification - $\chi_1(\tau)$ in the first function of the algorithm. The second function finds the best possible coalition that is suggested to the cooperators. If it gets rejected, the suggested coalition becomes a member of the set $\chi_1(\tau)$ and the second function is fired again (recursively). In general. we start with $\chi_1(\tau) = \{\}$.

### 6.4 Team Action Planning

Once a coalition is formed, the agents share a joint commitment to achieve the goal $\tau$. Within this phase, a team of collaborative agents jointly creates a team action plan $\pi(\tau)$. The team action plan, that is a result of the coalition planning activity, is a joint commitment structure that defines exactly how each team member will contribute to achieving the shared goal (amount of resources, deadlines, etc.). The coordinator is supposed to

(i) **decompose** a goal $\tau$ into subtasks $\{\tau_i\}$, and

(ii) **allocate** the subtasks within the already formed coalition $\chi(\tau)$.

There may be many achievable team action plans $\pi(\tau)$. The coordinator seeks for the cheapest or the fastest possible plan.

As there is no semi-private knowledge shared across the alliances, the agents from different alliances coordinate their activities by means of the contract net protocol. The intra-alliance team-action planning mechanism is not the pure acquaintance model contraction, where the team-

[2] Suggesting a possible coalition may be sometimes inherently complex. Design and implementation of intelligent algorithms for distributed coalition formation has been studied separately and their incorporation in CPlanT will be the subject of further research.

action plan would result from the coalition leader deliberation process followed by a contract. All the coalition members construct the precise team action plan collaboratively.

Let us view the team action plan as a resource allocation problem, where the coordinator is supposed to (i) decompose a goal $\tau$ into subtasks $\{\tau_i\}$ and (ii) allocate the subtasks within the coalition $\chi(\tau)$. There may be many team achievable action plans $\pi(\tau)$ – decompositions and subtasks' allocations within the coalition. In the team action planning we wanted the coalition members to find the optimal team action plan $\pi'(\tau)$ so that either

$$\forall\ \pi(\tau): \exists\ \pi'(\tau) : (\forall \pi(\tau) \neq \pi'(\tau), \forall \tau'_i \in\ \pi'(\tau) \wedge \forall \tau_i \in\ \pi(\tau): \max(\mathrm{due}(\tau'_i)) \leq \max(\mathrm{due}(\tau_i)))$$

or

$$\forall\ \pi(\tau): \exists\ \pi'(\tau) : (\forall \pi(\tau) \neq \pi'(\tau): \sum_{\tau_i \in \pi(\tau)} (\mathrm{cost}(\tau'_i)) \leq \sum_{\tau'_i \in \pi(\tau')} (\mathrm{cost}(\tau_i))).$$

The collaborators advertise their services in the most informative while efficient form. We have suggested the linear approximation of the discrete function that maps the delivery amount into due dates. Therefore, the coordinator's acquaintance model stores the social knowledge that is imprecise, but very compact and efficient to parse. According to this social knowledge, the coordinator suggests the most optimal request decomposition and resource allocation – $\pi(\tau)$ and transforms it into a contract proposal. This proposal is sent to the other coalition members, which reply with a specific collaboration proposal. However, the coordinator may find this proposal to be different than expected owing to the fact that the approximate information provided by the collaborator was far to imprecise. Instead of agreeing upon a joint commitment for this sub-optimal team action plan, the coordinator adapts the conflicting social knowledge and fires another round of negotiation. With each further negotiation stage, the team action plan should be closer to the optimal team action plan. This process is to be iterated until there is no conflict in the expected capacity of the collaborators and the proposed delivery.

The iterated team action planning negotiation protocol has been proposed, while just a single stage version of it has been implemented in the CPlanT multi-agent system. See Figure 7 for illustration of the task action planning negotiation from the view of the coordinator.

```
in idle times:
   recei□e-□e□□age(perform: request, sender: Ai, receiver: CL,
                        content: e□pected-time-price(τi))

in request time:
   re□eat
   □        □ec□□ □□□e (τ, social-knowledge) □ π(τ)
            ∀ π(τ)i ∈ π(τ)
            □en□-□e□□age(perform: a□□, sender: CL, receiver: Ai, content: π(τ)i)
            recei□e-□e□□age(perform: re□l□, receiver: CL sender: Ai, content: time-price(π(τ)i))
            if ∀ time-price(subtaski) (i□-cl□□e(time-price(π(τ)i), e□pected-time-price(π(τ)i))
                        then □en□-□e□□age(perform: request, sender: CL,  receiver: Ai, content: π(τ)i), return
                        else u□□ate(time-price(π(τ)i), e□pected-time-price(π(τ)i)
     □
       until false
```

**Figure □– Tea□ Acti□n Planning**

**Example:** Let us have two humanitarian relief providers: h-agent-1 and h-agent-2. The h-agent-1 can provide 3 rescuers in 1 day, 7 rescuers in 2 days and 10 rescuers in 3 days. The h-agent-2 is able to deliver 6 rescuers in the day one, 10 people in the middle of the second day, thirteen rescuers within 2 days and finally all 15 rescuers can be in place by three days. This information can be replaced by a simplified piece of the social knowledge in the form

`{(h-agent-1 10 3), (h-agent-2 15 3)}`

representing a linear approximation of the offered resources (See Figure 9).

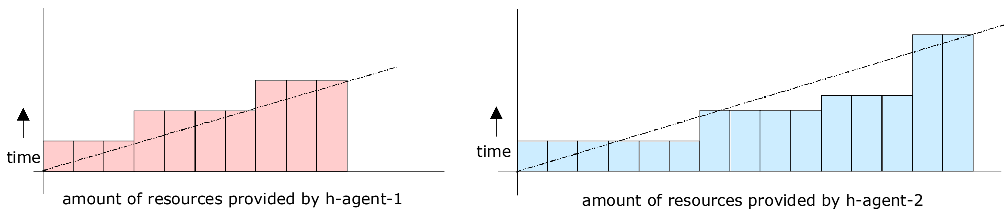


**Figure □– □inear a□□r□□□i□ati□n □□□□cial □n□□le□ge**

These two pieces of information can be put together and an optimal decomposition can be thus found (See Figure 9).

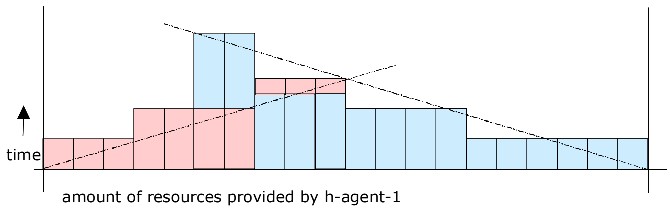


**Figure □– C□□□inati□n □□linear □□cial □n□□le□ge a□□r□□i□ati□n □r□□ t□□c□lla□□rat□r□**

However, if the social knowledge is not precise enough, the suggested decomposition may be far worse then expected (See Figure 10).

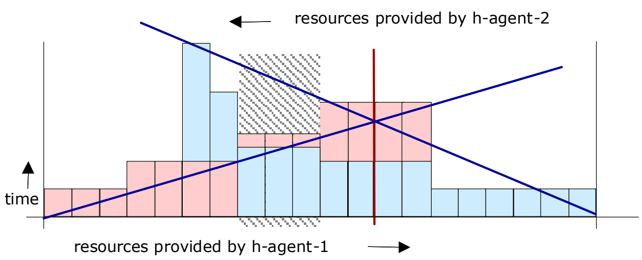


**Figure 1□– C□□□inati□n □□i□□reci□e linear □□cial □n□□le□ge a□□r□□i□ati□n □r□□ t□□ c□lla□□rat□r□**

In such a case the coordinator adapts its social knowledge according to these circumstances and suggest another decomposition that relies on more precise information (See Figure 11).

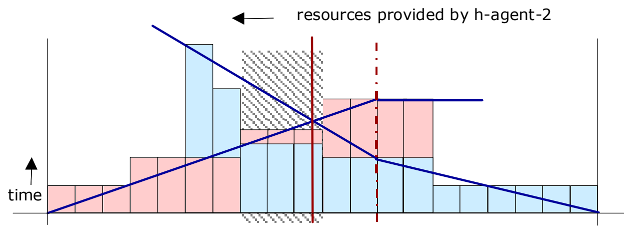


**Figure 11– C□□□inati□n □□a□a□te□linear a□□r□□i□ati□n □□□□cial □n□□le□ge □r□□ t□□ c□lla□□rat□r□**

# 7 Implementation and Testing

## 7.1 Implementation

Testing the correctness of the CPlanT required a well-defined, formal, but realistic enough scenario that can represent, model and initiate all aspects of agents' nontrivial behavior. The above specified principles and ideas have been tested and implemented on a subset of the OOTW types of operations – humanitarian relief operations. For this purpose we designed and implemented a hypothetical humanitarian scenario Sufferterra representing a suffering island and several imaginary countries ready to help. The Sufferterra scenario was inspired by [14], [15], [25]. The scenario knowledge has been encoded in XML and the computational model of the scenario has been implemented in Allegro Common Lisp.

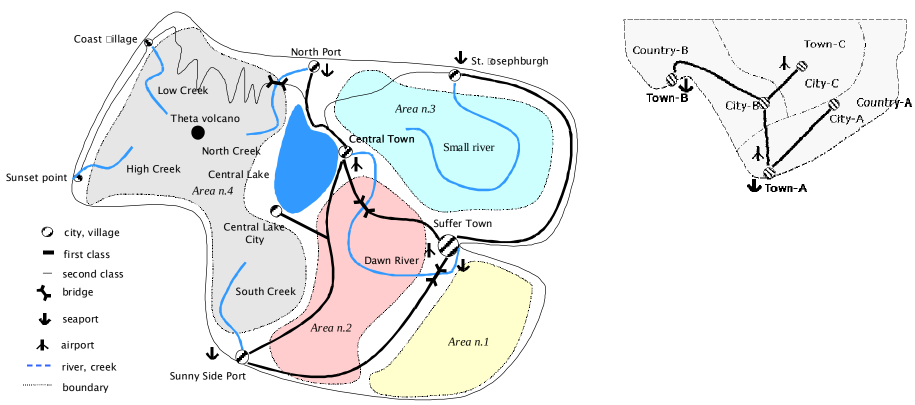


**Figure 1□– □u□□erterra – a general □cenari□□□r hu□anitarian □□erati□n□**

The R-Agents specify the physical arrangements of the geographical objects and the resources they provide. The problem specification does not distinguish the level of modeling granularity, i.e. each physical object may be implemented as an R-agent or several physical objects can make together an R-agent. For the testing purposes we have implemented a single R-Agent that represents the entire map of the area. The H-agents subscribe the R-Agent for specific information, by which these subscribers are informed about any change in physical arrangements of the relevant part of the map. There is a simple IN-Agent implemented as a part of the CPlanT community. Through one of the running instances of the IN-Agent, one can compose a "call-for-help" request and execute the coalition planning process. Such a request includes the type of disaster ("volcanic", "hurricane", "flood", "earthquake"), the degree of disaster (1..9), location and the targeted H–Agent.

```
<city>
      <name> "Suffer Town" </name>
      <national-composition> "((christian 67) (muslim 18) (native 13) (other 2))"
      </national-composition>
      <population> "50000" </population>
      <seaport>
             <ID> "1" </ID>
             <capacity> "25" </capacity>
             <material-hour> "200000" </material-hour>
      </seaport>
      <airport>
             <ID> "1" </ID>
             <capacity> "30" </capacity>
             <material-hour> "100000" </material-hour>
             <runway> "3000" </runway>
      </airport>
</city>
```

**Figure 1□– □□a□□le □□□M□□e□initi□n enc□□ing □□the □□u□□er T□□n□□□ect**

CPlanT has been successfully tested on the Sufferterra humanitarian relief scenario. The implementation is complemented by a visualizing meta-agent, which is implemented in Java. This meta-agent views logical structure of the system e.g. alliances, coalitions, team action plans and other properties of the community. There is a separate visualization for communication traffic monitoring. This component, that is not an agent, but rather a part of the multi-agent platform, serves mainly to debugging purposes. The community can be viewed and the requests can be sent from the web server via classical Internet browsers and from the WAP phones interface as well.

### 7.2 Experiments, Testing

The concept of the research has been tested on the CPlanT multi-agent system and the Sufferterra humanitarian relief scenario. The concept has been tested according to several different objectives:

- communication and computation requirements,
- quality of the solution provided,
- disclosure of private and semiprivate knowledge, and
- initialization phase of the community.

### 7.3 Communication requirements

An important part of the agent deliberation activities can be decomposed into inter-agent negotiation process. This is why we have concentrated our attention primarily to savings of the communication traffic in the entire system. The communication traffic has been observed in different architecture arrangements of the community (e.g. using different number of alliances) and for different complexity of the tasks (e.g. different number of contracts). Having 20 agents we have experimented with the sample of all agents being organized in one alliance, with agents clustered in 2, 4, 7 and 20 alliances. From the definition of the lifecycle of community follows that latter case ($\forall A$: $\mu(A)=\emptyset$) does not exploit any advantages of the acquaintance model contraction and the community behaves such as no social knowledge is administered and used. As the social knowledge requires lot of maintenance, we have also measured how does the maintenance messages affect the overall efficiency of the system.  All the experiments have been carried out on the set of 19 measurements for each community arrangement. The values in the graph are averages from the measurements. The detailed measurement results are recorded in Appendix.

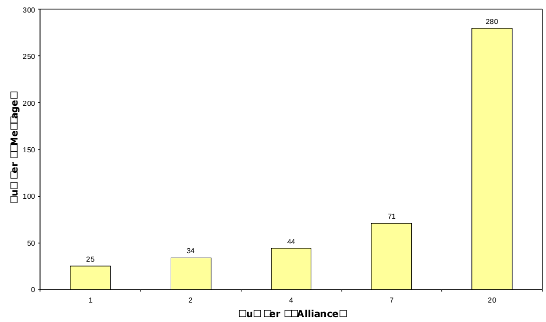


**Figure 1□ – C□□□unicati□n tra□□ic in critical ti□e□□□the c□□□unit□li□e**

As already explained, an important part of the communication traffic is carried out in the critical time – i.e. in the moment when the system is requested to provide a plan. By exploiting the social knowledge that has been prepared in advance, we aimed at minimising the communication traffic in this moment. The cost we have paid for this was the increased communication traffic in the idle times of the community. In the idle times the agents are busy with maintaining the social knowledge stored in their acquaintance models. The communication traffic grows with increasing the number of alliances as each alliance member stores more voluminous acquaintance model and each coalition leader searches for a coalition by parsing the acquaintance model only.

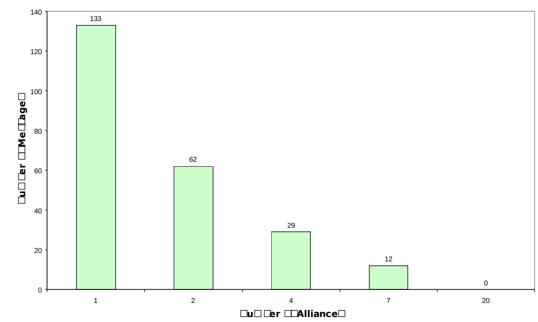


**Figure 1□- Ac□uaintance □□□el □aintenance- c□□□unicati□n in i□le ti□e□**

Figure 14 depicts dependency between the structure of the community (number of alliances) and the communication traffic in the critical times. Figure 15, on the other hand, illustrates dependency between the community architecture and the communication traffic required for the models' maintenance (with a larger social model we need more messages for maintaining the agent's acquaintance model).

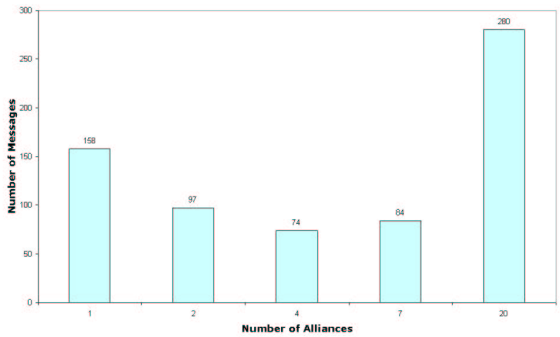


Figure 16 – Communication traffic in critical and idle time of the community

In Figure 16 there is the total communication traffic in the community depicted in the y axis. From these graphs we can see that with an increasing number of alliances, we can reduce the communication requirements for maintenance of the model while the most of the communication in the critical time we save when there is just one huge alliance. The optimal arrangement of the community with respect to the whole of inter-agent communication (both the critical time and the idle time – see Figure 16) was identified in the case of four alliances. However, it is not possible to define the optimal system structure because the agents cannot predict future tasks and the number of agents required for implementing these tasks. It is clear that for tasks requiring low number of agents we will prefer small alliances while for the tasks requiring many agents larger alliance will be preferred. It is necessary to be aware the fact that number of agents in one alliance must not be too high otherwise the maintaining of the large social model is too expensive from the communication point of view.

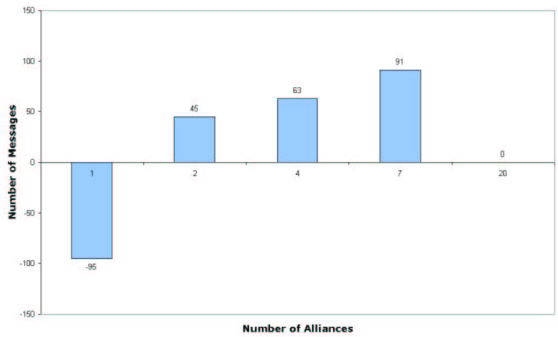


Figure 17 - Communication traffic reduction using acquaintance model

The Figure 17 shows how much of the communication traffic is saved by applying the acquaintance model (in comparison to the contract net protocol). In the case of 20 alliances (each agent being equipped with an acquaintance model) the exactly same behavior as in the contract net protocol processes is achieved. Each column in the graph in Figure 17 depicts how much we would save when not considering the acquaintance model but with the community structured in the system of alliances. In the case of one alliance, there is an intensive communication traffic required for the maintenance of the agents' acquaintance models. This is why the contract net protocol performs better in this singular case. The Figure 18 shows the extent of the communication traffic among the alliances.

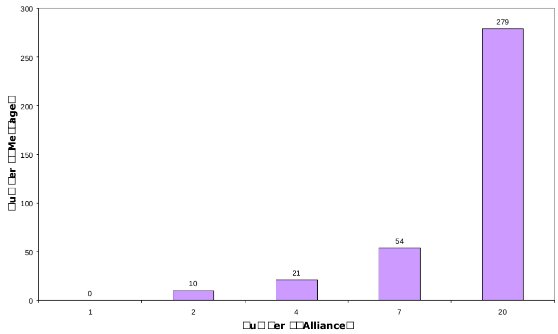


Figure 1□ – □□lu□e □□the inter alliance c□□□unicati□h

## 7.4 Quality of the coalition

The quality of the formed coalition (coalition value) is an important aspect in any coalition formation research. In the Sufferterra scenario, there are two key attributes that influences the coalition value:

– **success rate –** how many of the requested resources the coalition provides,

– **delivery time –** by when the coalition delivered the resources to the requestor.

The agents' preferences were set so that the agents try to maximize the success rate of the coalition. This is why the experiments resulted in the coalition with very similar success rates. We did not have any evidences to conclude any dependency between the success rate of the coalition and the used communication mechanism. However, with an increasing number of alliances in the system, the overall delivery time was kept increasing due to additional coordination costs among the alliances. The delivery time can be understood as an over-price for the humanitarian relief. As in the classical coalition research systems we can observe also in the CPlanT that with the increasing number of agents in a coalition the price increases (there is a so called coordination cost incorporated). Having singleton alliances, agents try to do most of the job by themselves. Doing so, they contribute to increase of their coalition value.

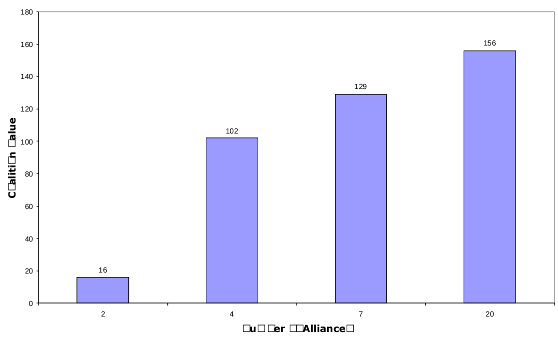


Figure 1□ – C□aliti□n □alue in□uence□□□the nu□□er □□alliance□

The graph in the Figure 19 depicts dependency of the coalition value with respect to different number of alliances. This relation very much depends on the delivery time that is depicted in the Figure 20. From this measurement we have confirmed our expectation that a high number of alliances results in better coalitions as there will be much less costs for coordinating the one-alliance-members within the coalition.

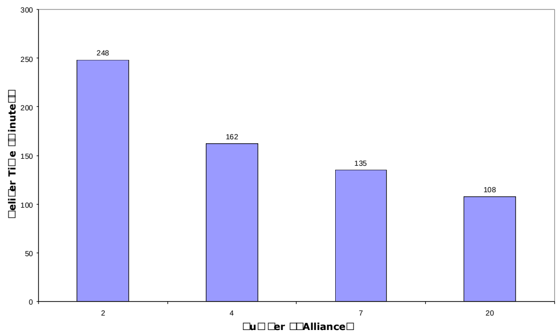


Figure □□ – □eli□er□ti□e

Figure 21 depicts the system response times. This measurement is closely related to the communication traffic in the community and verifies that the system responds quickly if there is a good structuring of the community (four alliances in our case).

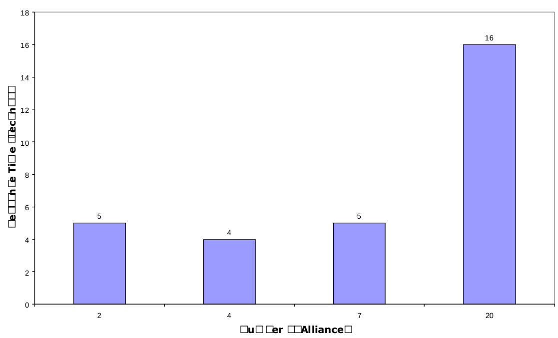


Figure □1 – □e□□□n□e ti□e□

### 7.5 Knowledge disclosure

The key challenge that our research addressed has been minimization of both the private and the semi-private knowledge disclosures. We have tried to measure both types of information disclosure. Once the private information is identified by another agent, this agent finds about the intent of the respective agent. This very often happens when an alliance fails to plan all the requests and starts a contract net protocol process within members of the other alliances. Those who will not be awarded the contract in the end know that the coordinator intends to operate in a mission where it needs the resources requested.

The semiprivate information is disclosed in the same situation, when the possible collaborator proposes a service (as a reaction to a coordinator call for collaboration) that will not be accepted by the coordinator. In such a case the coordinator finds out about the services the suggested collaborator can provide. Both the above-mentioned cases are classified as a strong knowledge disclosure, since they happen in the communication process among alliances. Weak knowledge disclosure happens in the registration phase within a single alliance and represents the amount of information that has become shared within the alliance. We have measured the knowledge disclosure as a number of elementary knowledge units representing intention to plan a mission, free allocation of a single resource, preference to (or not) form an alliance with an agent, etc.

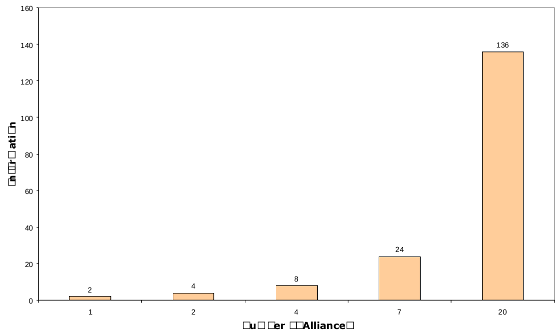


Figure □□ – □tr□ng □ri□ate in□□r□ati□n □i□cl□□ure

Figure 22 shows the dependence of the amount of the private information disclosure in different architectures of the community. As expected, the largest disclosure of intents comes about in the case of 20 alliances, as there is the highest CNP-based communication traffic among the alliances.

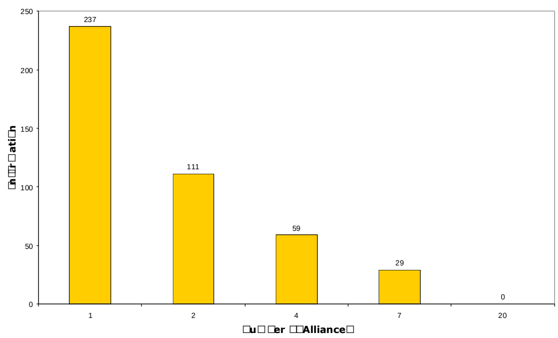


Figure □□ – □ea□ □e□i-□ri□ate in□□r□ati□n □i□cl□□ure

Figure 23 shows the weak disclosure of semi-private information that occurs in the alliance formation phase. There is not a weak disclosure once the agents are utterly independent (20 alliances) while important part of information is disclosed in the case of a single alliance. The weak disclosure of semi-private information is closely related to the acquaintance model maintenance.

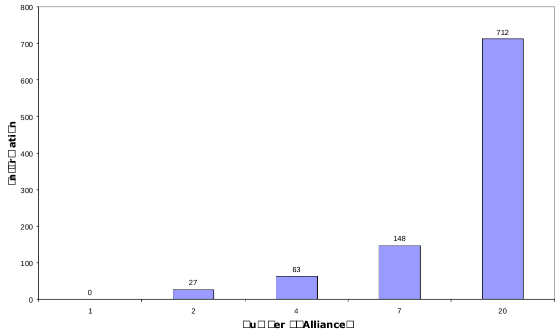


Figure □□– □tr□ng □e□i-□ri□ate in□□r□ati□n □i□cl□□ure

On the other hand, there is no strong semi-private information disclosure in one alliance while the independent agents are starting to loose their semi-private information in the strong sense. It makes no implication to put together the graphs in Figures 23 and 24 since it is hard to compare the significance of the weak and strong disclosures of knowledge.

An interesting fact is that neither of these two extreme cases (a community with a single alliance – $|\{\kappa_i\}| = 1$ and when each agent constitutes an own alliance – $|\{\kappa_i\}| = |\Theta|$) is the best for concealing the agents' private and semi-private knowledge. With one alliance, the semi-private knowledge becomes public while with no alliance each contract net protocol process will reveal information about the contractor intentions. It is rather difficult to find a good compromise in a number of alliances. What matters, is the probability that a request will not be fulfilled within one alliance and the coalition leader will have to subcontract other agents. The amount and structures of alliances in our domain emerge naturally according to the agents' private knowledge and other collaboration restrictions. Therefore it makes no sense to suggest an optimal number of alliances for a given community.

### 7.6 Initialisation of the community

The last set of measurements investigated behaviour of the community in the initialisation phase of the community life, where the alliance formation process plays an important role.

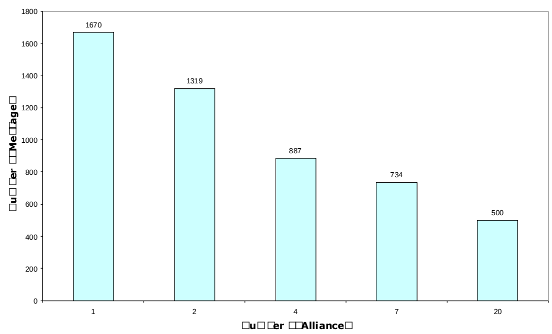


Figure □□– □nitiali□ati□n □ha□e

As shown in the Figure 25, with increasing independence of the agents the registration phase is easier. In the case of one alliance with lots of the shared semi-private knowledge, the initialization process becomes a bottleneck of the system's operation. The Figure 26 shows the semi-private information disclosure in the initialization phase.

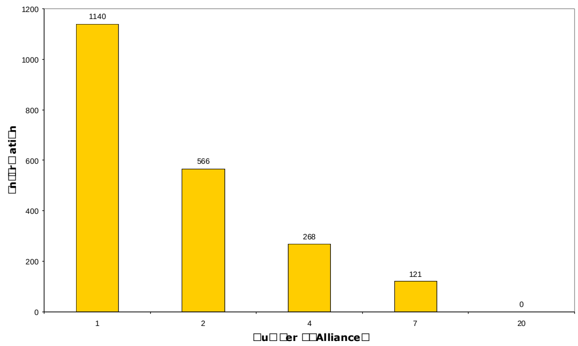


Figure □□– □e□i-□ri□ate in□□r□ati□n □□cl□□ure in the initiali□ati□n □ha□e

As mentioned earlier, the agents did not group into alliances in the most optimal way. With a different order of registration, various alliances may be closed. Apart from the range of services an alliance can provide, the amount of the disclosed private and semi-private knowledge is used for assessing appropriateness of an alliance allocation. Such a measure is planned to be used by a meta-agent who reasons on the top of the alliances and suggests promising alliance reconfigurations.

# 8 Relation to Coalition Planning Research

There has been a lot of work carried out in the area of coalition formation and coalition planning. It has been shown that finding the optimal coalition is a NP complete problem [17]. Researchers mainly suggest different negotiation strategies and analyze complexities of the coalition formation process [21]. When a subject of optimization is the quality of the formed coalition, the agents usually act **collaboratively**. There have been published many of centralized planning mechanisms for coalition formation [16], [17]. On the other hand, the **self-interested** agents maximize their own profit when participating in a coalition, no matter how well they will perform as a group. Many researchers analyzed properties of communities of self-interested agents such as their stability, worst-case profit, or payoff division among the agents [8]. The domain we have investigated is partially of cooperative and self-interested type at the same time. Humanitarian aid providing agents tend to cooperate in the time of a crisis while they are self-interested and compete each other in a long-term horizon. Therefore, there was suggested a concept of alliances – collectives of agents that agreed to collaborate (to potentially form a coalition).

More importantly, the profit is very often the key optimization criterion when the agents optimize a coalition formation process (either collaboratively or competing each other). Besides the quality of the coalition, in the OOTW domain there are two (maybe more important) aspects to be taken into account. As forming an optimal coalition is a very complex problem, the **response time** becomes an important issue. Agents are limited in resources and a reasonably good answer, that is quickly provided, is very often much better than an optimal coalition found later [20], [23]. Practitioners would add that implementing a multi-agent system with a large number of agents, that are supposed to interact heavily, results in a **communication traffic**

**overload** [5]. In our research, we have tried to decompose the reasoning process and distribute it among the agents. While keeping the agents' deliberation processes simple, we have concentrated our efforts on minimizing the communication interaction among the agents in order to suggest community structuring in a reasonable time. As the OOTW agents are also self-interested in certain way, they want to stay hidden in front of someone and advertise its collaborative capabilities to others. This is why we have to respect also the amount of **private information** to be disclosed. Therefore, we have also studied leaks of private information while forming the coalitions.

Research of the teamwork in a similar domain (evacuation scenarios) was reported in [24]. It was suggested to integrate the already existing software applications in the TEAMCORE wrapper agents. Unlike our acquaintance model that contains just social knowledge, the TEAMCORE wrapper agents also maintain the domain specific team plans and the hierarchy of goals. Teams of agents share a team-oriented program, which is a joint knowledge structure that coordinates their activities. In the CPlanT, there is no explicit team action plan distributed in agents' acquaintance models. The structure of the coalitions and the team-action plan is a result of the inter-agent negotiation process. However, combination of both approaches where the agents' behavior is coordinated by a team-action plan that results from the agents' negotiations seems to be an interesting topic for further research.

The investigators approaching the problem from the game-theoretic point of view solve the problem of a higher complexity. Whereas in our case, there is a hierarchy structure for each task that is sent to the community and each task is coordinated by a single agent (the coordinator), in [7] all agents are equal. The agents autonomously analyze their own value. Through negotiations, they try to find out which coalition is the most profitable for them to join. This problem is inherently more complex and causes communication problems in complex communities. There will be several stages of negotiations needed as in many cases optimality of cooperation between two agents may not be reciprocal. In our case, we did not need to solve such a complex problem. On the other hand, in the CPlanT we must optimize not only which coalition to join but also which services to provide to the coalition (e.g. team action planning). One may suggest that the game-theoretic approach could be used in the alliance formation phase of our algorithm. However, the agents join the system continuously, which makes it rather difficult to maintain the overall optimality of the distribution of alliances.

Besides the contract-net-protocol, there are other negotiation strategies based on classical auctioning mechanisms. While in combinatorial actions, the motivation of an agent is usually to make the biggest profit (or to contribute to a coalition in the best way), in our case, all the auctioneers and the bidding agents collaborate. A bidding agent tries to provide the auctioneer with such a bid that approximates in the best way the resources it can provide, and will help it to suggest the best possible resource allocation. In the CPlanT, the agents also do not speculate about whom to work with. As we optimize the private information losses, the collaboration within one alliance is always preferred. There is a potential of using the optimization for multiple auctioning mechanisms for the team action planning within several overlapping coalitions [1].

# 9 Conclusion

The research described in this paper contributes to the coalition formation community by suggesting an alternative, knowledge based approach to the problem. Our research has been driven by the very specific domain of the OOTW. Apart from the classical contract net protocol techniques, we have used the communication strategy based on combination of three techniques: the centralized registration, the acquaintance models and the contract net protocol negotiations.

The agents in the community are organized into smaller, disjunctive groups called alliances. Each agent in the alliance is able to start the negotiation process to form a coalition and to develop a team action plan for a specific task either within the alliance or in collaboration with other alliances. Inside-alliance negotiations explore mainly the social knowledge stored in the acquaintance models, but the CNP technique is used as well (especially in the phase of the team action planning). The inter-alliance negotiations are based just on the CNP principles.

The general complexity of negotiations, when forming a coalition in a MAS, is of an exponentially explosive nature [6], [16], [20]. It has been shown that finding and optimal coalition is an NP complete problem when no specific constraints are imposed. In our case, the negotiation complexity of the coalition formation problem has been significantly reduced because:

- agents are organized into several disjunctive sets (alliances) and the most of the coalitions are created just inside an alliance (i.e. a reduced space of negotiations is achieved)
- the coalition leader within an alliance is set randomly (each coalition member has got the same coordination capacity and can manage the negotiation process), they don't compete for the role.
- within an alliance, the negotiation process explores the acquaintance models (social knowledge) in combination with the CNP technique and the pure CNP negotiations are used just in the case of the inter-alliance negotiations.

While the contract net protocol runs rather inefficiently, it keeps the agents from different alliances independent (they do not have to disclose their semi-private knowledge across alliances). This is why, the acquaintance-model based planning has been used exclusively within the alliances.

In our approach, we have not prioritized the requirement for the global coalition optimality, as this is not the main challenge in the OOTW planning. The main issue has been to develop an acceptable plan without forcing the agencies (agents) to make their private knowledge (namely intents and resources) public. This quite specific OOTW requirement enabled to reduce the complexity of the negotiation problem significantly. It has been measured that optimality of the coalition value slightly increases with the number of alliances (the role of the acquaintance model is getting smaller), while the problem complexity with a smaller number of socially knowledgeable alliances is significantly reduced.

## References


[1] Anthony, P., Hall W., Dang, V., Jennings, N.R.: Autonomous Agents for Participating in Multiple On-line Auctions. In: Proc *IJCAI Workshop on E-Business and the Intelligent Web*, Seattle, WA, 2001, pp.54-64

[2] Barta, J., Pechoucek, M., Marik, V.: Sufferterra Humanitarian Crisis Scenario. *Project Report* of US Air Force Research Contract - Acquaintance Models in Operations Other Than War Coalition Formation, project contract no.: F61775-00-WE043, (US Air Force proprietary), Department of Cybernetics, Czech Technical University, January 2001.

[3] Dix J., Subrahmanian V.S., Pick G.: Meta Agent Programs. *Journal of Logic Programming*, 46(1-2):1-60, 2000

[4] Jennings, N. R.: Agent Based Computing: Promises and Perils (Computer and Thought Award Lecture). *International Conference on Artificial Intelligence*, Stockholm, 1999

[5] Kaminka, G. A., Pynadath, D. V., Tambe, M.: Monitoring Deployed Agent Teams. In: *Proceedings of the Fifth International Conference on Autonomous Agents* (Agents-2001), Montreal, Canada, 2001

[6] Ketchpel, S.: Coalition Formation Among Autonomous Agents. In: Proc. *5th European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, Neuchatel, Switzerland, 1993

[7] Klusch, M., Contreras, J., Wu, F., Shehory, O.: Coalition Formation in a Power Transmission Planning Environment. In: *Proc. 2nd International Conference on Practical Applications of Multi-Agent Systems PAAM-97*, London, UK, 1997

[8] Li, C., Sycara, K.: Algorithms for Combinatorial Coalition Formation and Payoff Division in an Electronic Marketplace. *Tech. Report CMU-RI-TR-01-33*, Robotics and Graduate School of Industrial Administration, Carnegie Mellon University, November, 2001.

[9] Nwana H.S., Ndumu D.T.: An Introduction to Agent Technology. In: *Software Agents and Softcomputing* (Nwana H.S. and Azarmi N. eds.), LNAI 1198, Springer-Verlag, Heidelberg, 1997, pp. 3-26

[10] Marik, V., Pechoucek, M., Stepankova, O., Lazansky, J.: ProPlanT: Multi-Agent System for Production Planning. *International Journal of Applied Artificial Intelligence*, 14:727-762, 2000

[11] Marik, V., Pechoucek, M., Stepankova, O.: Social Knowledge in Multi-Agent Systems, In: *Multi-Agent Systems and Applications* (Luck, M. et. al, eds.), LNAI 2086, Springer-Verlag, Heidelberg, 2001, pp. 211-245

[12] Pechoucek, M., Norrie, D.: Knowledge Structures for Reflective Multi-Agent Systems: On Reasoning About Other Agents. *Research Report No. 538*, Department of Mechanical and Manufacturing Engineering, University of Calgary, Alberta, Canada, December 2000

[13] Pechoucek, M., Marik, V., Stepankova O.: Towards Reducing Communication Traffic in Multi-Agent Systems. *Journal of Applied System Studies*, vol.2., No. 1, Cambridge International Science Publishing, Cambridge, UK, Spring, 2001.

[14] Rathmell, R. A.: A Coalition Force Scenario: Binni – Gateway to the Golden Bowl of Africa, *Defence Evaluation Research Agency*, 1999

[15] Reece, G. A., Tate, A.: The Pacifica NEO Scenario. *Technical Report ARPA-RL/O-Plan2/TR/3*. Artificial Intelligence Applications Institute, University of Edinburgh, Scotland, 1993

[16] Sandholm, T., Larson, K., Andersson, M., Shehory, O., Tohmé, F.: Coalition Structure Generation with Worst Case Guarantees. *Artificial Intelligence*, 111(1-2), 209-238, 1999

[17] Sandholm, T., Lesser, V.: Coalitions among Computationally Bounded Agents. *Artificial Intelligence 94(1)*, 99-137

[18] Sandholm, W.: Distributed Rational Decision Making. In: *Multi-Agent Systems: Modern Approach to Distributed Artificial Intelligence* (Weiss G., ed.), The MIT Press, Cambridge, 1999

[19] Shehory, O., Kraus S.: Formation of Overlapping Coalitions for Precedence Ordered Task-Execution Among the Autonomous Agents. In: *Proc. ICMAS-96*, Kyoto, Japan 1996, pp. 330-337

[20] Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence*, vol. 101 (1-2), pp. 165-200, 1998

[21] Shehory, O., Kraus, S.: Coalition Formation Among Autonomous Agents: Strategies and Complexity. In: *From Reaction to Cognition* (Castelfranchi, C., Muller, J.P. - eds.), LNAI No. 957, 1995, pp. 57-72

[22] Smith, R.: The Contract Net Protocol: High-level Communication and Control in Distributed Problem Solver. *IEEE Transactions on Computers*, 29(12):1104--1113, December 1980

[23] Steinmetz, J., Collins, M., Gini, M., Mobasher, B.: An Efficient Anytime Algorithm For Multiple-Component Bid Selection in Automated Contracting. In: *Agent Mediated Electronic Trading*, LNAI No. 1571, Springer-Verlag, Heidelberg, 1998, pp 105--125

[24] Tambe, M.: Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83 – 124, 1997.

[25] Walker, E. C. T.: Panel Report: Coalition Planning for Operations Other Than War. *Workshop at AIAI*, Edinburgh, Spring 1999

[26] Wellman, M.P.: A Market-Oriented Programming Environment and Its Applications to Distributed Multi-Commodity Flow Problems. *Journal of Artificial Intelligence Research*, 1 1-23, 1993

# Appendix

In the Appendix, there are experimental results presented in a tabular form. These results have been aggregated into graphs (see Section 7). We carried out 19 measurements for five different global system structures (different number of alliances). Each measurement consists of three phases according to the degree of the considered disaster. The task receiver (the future coordinator of the task), the type of the disaster and its location have been chosen randomly. Each measurement consists of two parts: **the task solution evaluation** – the first table and **the evaluation of communication** – the second table.

Description of labels used for the description of the measured values and results:

- **C□□ering** – to which extent the requirements have been covered,
- **□eli□er** – the time in minutes for delivering of the task solution, it includes the coordination and travel times as well,
- **□e□□□n□e** – activities planning response time,
- **□□□** – number of participating organizations,
- **A** – number of contract type messages,
- **M** – number of maintenance messages,
- **C□P** – number of additional messages the agents sent (without using a social model),
- **A□M** – number of all messages needed for all activities planning,
- **A□C□P** – number of all messages for all activities planning (without any social model to be used),
- **□ - □A□M□- □A□M□**– communication savings,
- **□A** – number of inter-alliance messages,
- **P** – private information disclosures, measured in the form of a number elementary knowledge units disclosed
- **□P□** – semi-private weak information disclosures
- **□P□**- semi-private strong information disclosures

Each first row in the communication evaluation table provides values that were acquired in the period of the community initialisation. The values that were not directly measured, but computed are denoted by a asterisks.

**Alliance□– □□ □Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| 1 | C-□CO | Hurricane | 2 | City-A | 100 | 101 | 21 | 7 |
| □ | C-C G | Flood | 2 | Central Town | 100 | 132 | 19 | 8 |
| □ | C-□HO | Hurricane | 2 | Coast Village | 100 | 101 | 10 | 4 |
| □ | ST G | Earthquake | 2 | Town-□ | 84 | 143 | 35 | 13 |
| □ | C-A A | Flood | 2 | Suffer Town | 44 | 101 | 9 | 3 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P□M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 500 | 0 | 0 | 500 | 500 | 0 | 0 | 0 | 0 | 0 |
| 1 | 283 | 0 | 0 | 283 | 283 | 0 | 282 | 136 | 0 | 501 |
| □ | 321 | 0 | 0 | 321 | 321 | 0 | 320 | 154 | 0 | 661 |
| □ | 123 | 0 | 0 | 123 | 123 | 0 | 122 | 58 | 0 | 237 |
| □ | 559 | 0 | 0 | 559 | 559 | 0 | 558 | 268 | 0 | 1374 |
| □ | 159 | 0 | 0 | 159 | 159 | 0 | 158 | 78 | 0 | 480 |

**Alliance□– □□ □Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| 1 | C-A G | Volcanic | 5 | Sunny Side Port | 100 | 127 | 18 | 8 |
| □ | C-□G | Volcanic | 5 | Suffer Town | 100 | 131 | 19 | 10 |
| □ | C-C G | Hurricane | 5 | North Port | 81 | 112 | 14 | 6 |
| □ | ST G | Hurricane | 5 | Sunset Point | 57 | 203 | 9 | 3 |
| □ | C-□Army | Flood | 5 | Town-A | 61 | 69 | 21 | 10 |
| □ | ST Police | Flood | 5 | City-□ | 38 | 92 | 14 | 6 |
| □ | C-A G | Earthquake | 5 | Town-C | 22 | 57 | 8 | 3 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P□M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 500 | 0 | 0 | 500 | 500 | 0 | 0 | 0 | 0 | 0 |
| 1 | 321 | 0 | 0 | 321 | 321 | 0 | 320 | 169 | 0 | 559 |
| □ | 401 | 0 | 0 | 401 | 401 | 0 | 400 | 192 | 0 | 797 |
| □ | 279 | 0 | 0 | 279 | 279 | 0 | 278 | 135 | 0 | 639 |
| □ | 119 | 0 | 0 | 119 | 119 | 0 | 118 | 57 | 0 | 303 |
| □ | 439 | 0 | 0 | 439 | 439 | 0 | 438 | 211 | 0 | 1298 |
| □ | 279 | 0 | 0 | 279 | 279 | 0 | 278 | 133 | 0 | 586 |
| □ | 159 | 0 | 0 | 159 | 159 | 0 | 158 | 84 | 0 | 586 |

| Alliance□– □□ □Ta□□ □egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □** | **□eli□er □□ □** | **□e□□□n□e □□□** | **□□□ □□u□ □er□** |
| **1** | C-A G | Earthquake | 7 | St. □osephburg | 100 | 162 | 20 | 6 |
| □ | C-C MO | Hurricane | 7 | Suffer Town | 99 | 131 | 26 | 12 |
| □ | C-□HO | Volcanic | 7 | City-□ | 68 | 104 | 26 | 10 |
| □ | CH ST HO | Flood | 7 | Town-□ | 46 | 131 | 28 | 11 |
| □ | ST G | Flood | 7 | Sunny Side Port | 22 | 27 | 3 | 1 |
| □ | C-□G | Hurricane | 7 | Town-A | 25 | 69 | 5 | 1 |
| □ | M C-□HO | Earthquake | 7 | Town-C | 22 | 75 | 9 | 3 |

| **Ta□□** | **A □M□** | **M □M□** | **C□P □M□□** | **A□M□M□** | **A□C□P□M□□** | **□□M□□** | **□A □M□** | **P □□□** | **□P□ □□□** | **□P□□□□** |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 500 | 0 | 0 | 500 | 500 | 0 | 0 | 0 | 0 | 0 |
| **1** | 241 | 0 | 0 | 241 | 241 | 0 | 240 | 114 | 0 | 543 |
| □ | 481 | 0 | 0 | 481 | 481 | 0 | 480 | 231 | 0 | 939 |
| □ | 439 | 0 | 0 | 439 | 439 | 0 | 438 | 213 | 0 | 1106 |
| □ | 441 | 0 | 0 | 441 | 441 | 0 | 440 | 212 | 0 | 1859 |
| □ | 39 | 0 | 0 | 39 | 39 | 0 | 38 | 21 | 0 | 159 |
| □ | 79 | 0 | 0 | 79 | 79 | 0 | 78 | 38 | 0 | 172 |
| □ | 161 | 0 | 0 | 161 | 161 | 0 | 160 | 85 | 0 | 737 |

| Alliance□– □□Ta□□ □egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □** | **□eli□er □□ □** | **□e□□□n□e □□□** | **□□□ □□u□ □er□** |
| **1** | C-□CO | Hurricane | 2 | City-A | 100 | 135 | 7 | 10 |
| □ | C-C G | Flood | 2 | Central Town | 100 | 192 | 6 | 9 |
| □ | C-□HO | Hurricane | 2 | Coast Village | 100 | 99 | 3 | 4 |
| □ | ST G | Earthquake | 2 | Town-□ | 84 | 161 | 11 | 13 |
| □ | C-A A | Flood | 2 | Suffer Town | 44 | 152 | 4 | 5 |

| **Ta□□** | **A □M□** | **M □M□** | **C□P □M□□** | **A□M□M□** | **A□C□P□M□□** | **□□M□□** | **□A □M□** | **P □□□** | **□P□ □□□** | **□P□□□□** |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 658 | 76 | 0 | 734 | 658 | -76 | 0 | 86 | 121 | 0 |
| **1** | 69 | 18 | 82 | 87 | 151 | 64 | 44 | 19 | 46 | 88 |
| □ | 77 | 14 | 108 | 91 | 185 | 94 | 56 | 24 | 46 | 96 |
| □ | 25 | 8 | 30 | 33 | 55 | 22 | 16 | 7 | 19 | 20 |
| □ | 125 | 17 | 182 | 142 | 307 | 165 | 96 | 42 | 47 | 237 |
| □ | 53 | 8 | 78 | 61 | 131 | 70 | 40 | 18 | 17 | 121 |

**Alliance□– □□Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| **1** | C-A G | Volcanic | 5 | Sunny Side Port | 100 | 217 | 7 | 9 |
| □ | C-□G | Volcanic | 5 | Suffer Town | 100 | 161 | 6 | 11 |
| □ | C-C G | Hurricane | 5 | North Port | 81 | 146 | 7 | 10 |
| □ | ST G | Hurricane | 5 | Sunset Point | 57 | 142 | 2 | 1 |
| □ | C-□Army | Flood | 5 | Town-A | 61 | 126 | 10 | 9 |
| □ | ST Police | Flood | 5 | City-□ | 38 | 122 | 8 | 13 |
| □ | C-A G | Earthquake | 5 | Town-C | 22 | 105 | 4 | 4 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P□M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□ □□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 658 | 76 | 0 | 734 | 658 | -76 | 0 | 86 | 121 | 0 |
| **1** | 82 | 14 | 112 | 96 | 194 | 98 | 56 | 25 | 37 | 80 |
| □ | 85 | 18 | 108 | 103 | 193 | 90 | 56 | 24 | 45 | 94 |
| □ | 93 | 18 | 130 | 111 | 223 | 112 | 68 | 30 | 38 | 155 |
| □ | 13 | 2 | 26 | 15 | 39 | 24 | 12 | 6 | 8 | 27 |
| □ | 109 | 15 | 182 | 124 | 291 | 167 | 96 | 42 | 28 | 313 |
| □ | 123 | 17 | 182 | 140 | 305 | 165 | 94 | 42 | 27 | 345 |
| □ | 57 | 7 | 104 | 64 | 161 | 97 | 52 | 24 | 11 | 206 |

**Alliance□– □□Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| **1** | C-A G | Earthquake | 7 | St. □osephburg | 100 | 222 | 6 | 7 |
| □ | C-C MO | Hurricane | 7 | Suffer Town | 99 | 161 | 6 | 11 |
| □ | C-□HO | Volcanic | 7 | City-□ | 68 | 86 | 7 | 10 |
| □ | CH ST HO | Flood | 7 | Town-□ | 46 | 154 | 7 | 12 |
| □ | ST G | Flood | 7 | Sunny Side Port | 22 | 27 | 5 | 1 |
| □ | C-□G | Hurricane | 7 | Town-A | 25 | 69 | 1 | 1 |
| □ | M C-□HO | Earthquake | 7 | Town-C | 22 | 105 | 4 | 4 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P □M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□ □□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 658 | 76 | 0 | 734 | 658 | -76 | 0 | 0 | 121 | 0 |
| **1** | 59 | 14 | 82 | 73 | 141 | 68 | 42 | 18 | 43 | 69 |
| □ | 85 | 20 | 104 | 105 | 189 | 84 | 56 | 25 | 50 | 118 |
| □ | 93 | 18 | 130 | 111 | 223 | 112 | 68 | 30 | 38 | 182 |
| □ | 113 | 20 | 156 | 133 | 269 | 136 | 84 | 37 | 39 | 339 |
| □ | 13 | 2 | 26 | 15 | 39 | 24 | 12 | 6 | 4 | 55 |
| □ | 27 | 2 | 52 | 29 | 79 | 50 | 26 | 12 | 4 | 68 |
| □ | 61 | 6 | 104 | 67 | 165 | 98 | 56 | 25 | 7 | 215 |

| Alliance□– □□Ta□□□egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □** | **□eli□er □□□** | **□e□□□n□e □□□** | **□□□ □□u□ □er□** |
| **1** | C-□CO | Hurricane | 2 | City-A | 100 | 182 | 5 | 8 |
| □ | C-C G | Flood | 2 | Central Town | 100 | 267 | 5 | 11 |
| □ | C-□HO | Hurricane | 2 | Coast Village | 100 | 203 | 6 | 5 |
| □ | ST G | Earthquake | 2 | Town-□ | 84 | 214 | 6 | 12 |
| □ | C-A A | Flood | 2 | Suffer Town | 44 | 206 | 3 | 4 |

| **Ta□□** | **A □M□** | **M □M□** | **C□P □M□□** | **A□M□M□** | **A□C□P □M□□** | **□□M□□** | **□A □M□** | **P □□□** | **□P□ □□□** | **□P□□□□** |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 715 | 172 | 0 | 887 | 715 | -172 | 0 | 88 | 268 | 0 |
| **1** | 35 | 38 | 40 | 73 | 75 | 2 | 10 | 4 | 83 | 11 |
| □ | 57 | 39 | 106 | 96 | 163 | 67 | 24 | 9 | 94 | 42 |
| □ | 23 | 14 | 42 | 37 | 65 | 28 | 10 | 4 | 26 | 11 |
| □ | 67 | 46 | 128 | 113 | 195 | 82 | 30 | 12 | 114 | 82 |
| □ | 31 | 13 | 96 | 44 | 127 | 83 | 22 | 9 | 25 | 76 |

| Alliance□– □□Ta□□□egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □** | **□eli□er □□□** | **□e□□□n□e □□□** | **□□□ □□u□ □er□** |
| **1** | C-A G | Volcanic | 5 | Sunny Side Port | 100 | 277 | 6 | 9 |
| □ | C-□G | Volcanic | 5 | Suffer Town | 100 | 266 | 6 | 12 |
| □ | C-C G | Hurricane | 5 | North Port | 81 | 251 | 6 | 12 |
| □ | ST G | Hurricane | 5 | Sunset Point | 57 | 142 | 1 | 1 |
| □ | C-□Army | Flood | 5 | Town-A | 61 | 56 | 6 | 11 |
| □ | ST Police | Flood | 5 | City-□ | 38 | 146 | 5 | 10 |
| □ | C-A G | Earthquake | 5 | Town-C | 22 | 32 | 4 | 6 |

| **Ta□□** | **A □M□** | **M □M□** | **C□P □M□□** | **A□M□M□** | **A□C□P □M□□** | **□□M□□** | **□A □M□** | **P □□□** | **□P□ □□□** | **□P□□□□** |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 715 | 172 | 0 | 887 | 715 | -172 | 0 | 88 | 268 | 0 |
| **1** | 43 | 40 | 74 | 83 | 117 | 34 | 14 | 6 | 93 | 18 |
| □ | 59 | 40 | 100 | 99 | 159 | 60 | 22 | 9 | 85 | 29 |
| □ | 65 | 43 | 128 | 108 | 193 | 85 | 28 | 12 | 72 | 58 |
| □ | 7 | 2 | 32 | 9 | 39 | 30 | 6 | 3 | 8 | 15 |
| □ | 67 | 38 | 160 | 105 | 227 | 122 | 38 | 15 | 51 | 128 |
| □ | 59 | 46 | 128 | 105 | 187 | 82 | 30 | 12 | 74 | 119 |
| □ | 43 | 18 | 128 | 61 | 171 | 110 | 30 | 12 | 27 | 138 |

**Alliance□– □□Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| 1 | C-A G | Earthquake | 7 | St. □osephburg | 100 | 282 | 8 | 8 |
| □ | C-C MO | Hurricane | 7 | Suffer Town | 99 | 266 | 6 | 12 |
| □ | C-□HO | Volcanic | 7 | City-□ | 68 | 84 | 7 | 11 |
| □ | CH ST HO | Flood | 7 | Town-□ | 46 | 21 | 6 | 11 |
| □ | ST G | Flood | 7 | Sunny Side Port | 22 | 27 | 1 | 1 |
| □ | C-□G | Hurricane | 7 | Town-A | 25 | 69 | 1 | 1 |
| □ | M C-□HO | Earthquake | 7 | Town-C | 22 | 87 | 3 | 5 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P □M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 715 | 172 | 0 | 887 | 715 | -172 | 0 | 88 | 268 | 0 |
| 1 | 39 | 35 | 74 | 74 | 113 | 39 | 14 | 6 | 99 | 10 |
| □ | 61 | 39 | 96 | 100 | 157 | 57 | 24 | 10 | 84 | 51 |
| □ | 63 | 52 | 128 | 115 | 191 | 76 | 30 | 12 | 86 | 76 |
| □ | 57 | 49 | 96 | 106 | 153 | 47 | 24 | 10 | 79 | 115 |
| □ | 7 | 2 | 32 | 9 | 39 | 30 | 6 | 3 | 4 | 31 |
| □ | 15 | 2 | 64 | 17 | 79 | 62 | 14 | 6 | 4 | 44 |
| □ | 41 | 11 | 128 | 52 | 169 | 117 | 32 | 13 | 13 | 161 |

**Alliance□– □□Ta□□ □egree – □**

| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□ □ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
|---|---|---|---|---|---|---|---|---|
| 1 | C-□CO | Hurricane | 2 | City-A | 100 | 307 | 5 | 7 |
| □ | C-C G | Flood | 2 | Central Town | 100 | 387 | 8 | 11 |
| □ | C-□HO | Hurricane | 2 | Coast Village | 100 | 259 | 4 | 6 |
| □ | ST G | Earthquake | 2 | Town-□ | 84 | 278 | 8 | 12 |
| □ | C-A A | Flood | 2 | Suffer Town | 44 | 266 | 3 | 5 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P □M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 955 | 364 | 0 | 1319 | 955 | -364 | 0 | 187 | 566 | 0 |
| 1 | 27 | 55 | 16 | 82 | 43 | -39 | 2 | 1 | 120 | 0 |
| □ | 47 | 96 | 120 | 143 | 167 | 24 | 10 | 4 | 179 | 7 |
| □ | 25 | 40 | 76 | 65 | 101 | 36 | 8 | 3 | 77 | 5 |
| □ | 57 | 97 | 164 | 154 | 221 | 67 | 16 | 6 | 174 | 45 |
| □ | 27 | 48 | 108 | 75 | 135 | 60 | 10 | 4 | 72 | 33 |

| Alliance□– □□Ta□□□egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□□ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
| 1 | C-A G | Volcanic | 5 | Sunny Side Port | 100 | 382 | 8 | 8 |
| □ | C-□G | Volcanic | 5 | Suffer Town | 100 | 326 | 5 | 10 |
| □ | C-C G | Hurricane | 5 | North Port | 81 | 323 | 6 | 10 |
| □ | ST G | Hurricane | 5 | Sunset Point | 57 | 42 | 2 | 1 |
| □ | C-□Army | Flood | 5 | Town-A | 61 | 276 | 6 | 11 |
| □ | ST Police | Flood | 5 | City-□ | 38 | 279 | 6 | 9 |
| □ | C-A G | Earthquake | 5 | Town-C | 22 | 135 | 2 | 3 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P□M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□ □□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 955 | 364 | 0 | 1319 | 955 | -364 | 0 | 187 | 566 | 0 |
| 1 | 35 | 78 | 76 | 113 | 111 | -2 | 6 | 2 | 157 | 9 |
| □ | 45 | 71 | 128 | 116 | 173 | 57 | 12 | 4 | 147 | 14 |
| □ | 47 | 79 | 160 | 126 | 207 | 81 | 14 | 6 | 116 | 27 |
| □ | 5 | 8 | 56 | 13 | 61 | 48 | 4 | 2 | 32 | 11 |
| □ | 53 | 95 | 160 | 148 | 213 | 65 | 16 | 6 | 128 | 56 |
| □ | 45 | 84 | 164 | 129 | 209 | 80 | 16 | 6 | 143 | 66 |
| □ | 17 | 24 | 112 | 41 | 129 | 88 | 8 | 4 | 44 | 25 |

| Alliance□– □□Ta□□□egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □ | □eli□er □□□ | □e□□□n□e □□□ | □□□ □□u□ □er□ |
| 1 | C-A G | Earthquake | 7 | St. □osephburg | 100 | 327 | 12 | 9 |
| □ | C-C MO | Hurricane | 7 | Suffer Town | 99 | 356 | 9 | 16 |
| □ | C-□HO | Volcanic | 7 | City-□ | 68 | 249 | 5 | 8 |
| □ | CH ST HO | Flood | 7 | Town-□ | 46 | 327 | 10 | 10 |
| □ | ST G | Flood | 7 | Sunny Side Port | 22 | 27 | 1 | 1 |
| □ | C-□G | Hurricane | 7 | Town-A | 25 | 69 | 1 | 1 |
| □ | M C-□HO | Earthquake | 7 | Town-C | 22 | 104 | 2 | 4 |

| Ta□□ | A □M□ | M □M□ | C□P □M□□ | A□M□M□ | A□C□P□M□□ | □□M□□ | □A □M□ | P □□□ | □P□ □□□ | □P□ □□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 955 | 364 | 0 | 1319 | 955 | -364 | 0 | 187 | 566 | 0 |
| 1 | 39 | 65 | 132 | 104 | 171 | 67 | 10 | 4 | 151 | 7 |
| □ | 69 | 137 | 108 | 206 | 177 | -29 | 12 | 5 | 241 | 26 |
| □ | 37 | 78 | 104 | 115 | 141 | 26 | 8 | 4 | 136 | 13 |
| □ | 45 | 94 | 108 | 139 | 153 | 14 | 12 | 5 | 144 | 56 |
| □ | 5 | 8 | 56 | 13 | 61 | 48 | 4 | 2 | 16 | 23 |
| □ | 11 | 8 | 108 | 19 | 119 | 100 | 10 | 4 | 16 | 36 |
| □ | 21 | 26 | 108 | 47 | 129 | 82 | 12 | 5 | 28 | 59 |

| Alliance□– 1 □Ta□□ □egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □□** | **□eli□er □□ □□** | **□e□□□n□e □□□□** | **□□□ □□u□ □er□□** |
| **1** | C-□CO | Hurricane | 2 | City-A | 100 | □ | high | 8 |
| □ | C-C G | Flood | 2 | Central Town | 100 | □ | high | 9 |
| □ | C-□HO | Hurricane | 2 | Coast Village | 100 | □ | high | 4 |
| □ | ST G | Earthquake | 2 | Town-□ | 84 | □ | high | 12 |
| □ | C-A A | Flood | 2 | Suffer Town | 44 | □ | high | 4 |

| Ta□□ | A □M□□ | M □M□□ | C□P □M□□ | A□M□M□□ | A□C□P□M□□ | □□M□□ | □A □M□□ | P □□□□ | □P□ □□□□ | □P□ □□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 880 | 560 | 0 | 1440 | 880 | -560 | 0 | 380 | high | |
| **1** | 31 | 152 | 38 | 183 | 69 | -114 | 0 | very low | high | 0 |
| □ | 33 | 171 | 38 | 204 | 71 | -133 | 0 | very low | high | 0 |
| □ | 15 | 76 | 38 | 91 | 53 | -38 | 0 | very low | high | 0 |
| □ | 45 | 228 | 38 | 273 | 83 | -190 | 0 | very low | high | 0 |
| □ | 13 | 76 | 38 | 89 | 51 | -38 | 0 | very low | high | 0 |

| Alliance□– 1 □Ta□□ □egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Ta□□** | **□ecei□er** | **□i□a□ter** | **□egree** | **□□cati□n** | **C□□ering □□ □□** | **□eli□er □□ □□** | **□e□□□n□e □□□□** | **□□□ □□u□ □er□□** |
| **1** | C-A G | Volcanic | 5 | Sunny Side Port | 100 | □ | high | 8 |
| □ | C-□G | Volcanic | 5 | Suffer Town | 100 | □ | high | 10 |
| □ | C-C G | Hurricane | 5 | North Port | 81 | □ | high | 9 |
| □ | ST G | Hurricane | 5 | Sunset Point | 57 | □ | high | 1 |
| □ | C-□Army | Flood | 5 | Town-A | 61 | □ | high | 10 |
| □ | ST Police | Flood | 5 | City-□ | 38 | □ | high | 9 |
| □ | C-A G | Earthquake | 5 | Town-C | 22 | □ | high | 4 |

| Ta□□ | A □M□□ | M □M□□ | C□P □M□□ | A□M□M□□ | A□C□P□M□□ | □□M□□ | □A □M□□ | P □□□□ | □P□ □□□□ | □P□ □□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| **□tart** | 880 | 560 | 0 | 1440 | 880 | -560 | 0 | 380 | high | 0 |
| **1** | 29 | 152 | 38 | 181 | 67 | -114 | 0 | very low | high | 0 |
| □ | 37 | 190 | 38 | 227 | 75 | -152 | 0 | very low | high | 0 |
| □ | 33 | 171 | 38 | 204 | 71 | -133 | 0 | very low | high | 0 |
| □ | 1 | 19 | 38 | 20 | 39 | 19 | 0 | very low | high | 0 |
| □ | 37 | 190 | 38 | 227 | 75 | -152 | 0 | very low | high | 0 |
| □ | 33 | 171 | 38 | 204 | 71 | -133 | 0 | very low | high | 0 |
| □ | 13 | 76 | 38 | 89 | 51 | -38 | 0 | very low | high | 0 |

| Alliance□– 1 □Ta□□ □egree – □ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Ta□□ | □ecei□er | □i□a□ter | □egree | □□cati□n | C□□ering □□ □□ | □eli□er □□ □□ | □e□□□n□e □□□□ | □□□ □□u□ □er□□ |
| 1 | C-A G | Earthquake | 7 | St. □osephburg | 100 | □ | high | 7 |
| □ | C-C MO | Hurricane | 7 | Suffer Town | 99 | □ | high | 12 |
| □ | C-□HO | Volcanic | 7 | City-□ | 68 | □ | high | 9 |
| □ | CH ST HO | Flood | 7 | Town-□ | 46 | □ | high | 11 |
| □ | ST G | Flood | 7 | Sunny Side Port | 22 | □ | high | 1 |
| □ | C-□G | Hurricane | 7 | Town-A | 25 | □ | high | 1 |
| □ | M C-□HO | Earthquake | 7 | Town-C | 22 | □ | high | 4 |

| Ta□□ | A □M□□ | M □M□□ | C□P □M□□ | A□M□M□□ | A□C□P □M□□ | □□M□□ | □A □M□□ | P □□□□ | □P□ □□□□ | □P□ □□□□ |
|---|---|---|---|---|---|---|---|---|---|---|
| □tart | 880 | 560 | 0 | 1440 | 880 | -560 | 0 | 380 | high | 0 |
| 1 | 25 | 133 | 38 | 158 | 63 | -95 | 0 | very low | high | 0 |
| □ | 47 | 228 | 38 | 275 | 85 | -190 | 0 | very low | high | 0 |
| □ | 33 | 171 | 38 | 204 | 71 | -133 | 0 | very low | high | 0 |
| □ | 43 | 209 | 38 | 252 | 81 | -171 | 0 | very low | high | 0 |
| □ | 1 | 19 | 38 | 20 | 39 | 19 | 0 | very low | high | 0 |
| □ | 1 | 19 | 38 | 20 | 39 | 19 | 0 | very low | high | 0 |
| □ | 15 | 76 | 38 | 91 | 53 | -38 | 0 | very low | high | 0 |