Design and Simulation of Trust Management Techniques for a Coalition Data Sharing Environment

Dr.Bhavani Thuraisingham

Srinivasan Iyer

The University of Texas at Dallas

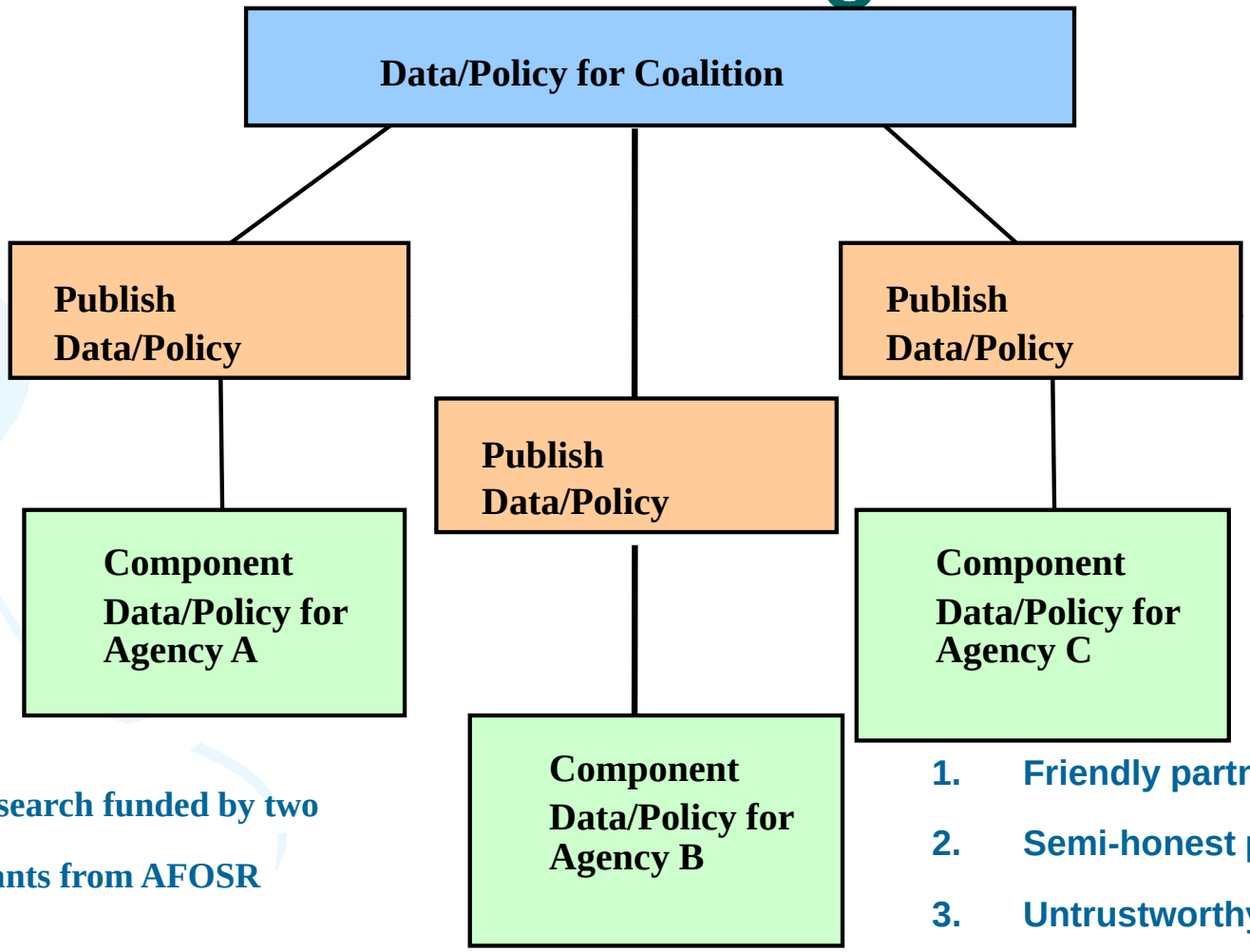
Objectives

- Our objectives are to simulate a Multi agent system for information sharing with confidentiality and trust Management.
- Calculate the amount of data lost and gained by each of the agent.

Outline of the presentation

- Vision
- Introduction
- Definition of an agent
- Functions of an agent
- System Architecture
- Algorithm for Information sharing within confidentiality and trust management
- Issues in Information sharing
- Simulation
- Conclusion

Vision: Assured Information Sharing



Research funded by two grants from AFOSR

1. Friendly partners
2. Semi-honest partners
3. Untrustworthy partners

Introduction

- Exciting emerging applications require hundreds or thousands of agents and robots to coordinate to achieve their joint goals.
- In domains such as military operations, space or disaster response, coordination among large numbers of agents promises to revolutionize the effectiveness of our ability to achieve complex goals.

Introduction (Contd..)

- Issues in such domains are
- Limited communication channels
- No agent can know the entire View of the environment
- When agent has some information, Should decide who to send it.
- Broadcasting is not desirable

Introduction (Contd..)

- Example scenario
- A Field agent in a military assignment on his route to an operation, observes many battlefield features. It needs to decide which combat group would need the info.

Introduction

- This project deals with the following
- How to find which agent needs what information?
- Is the source agent trustworthy?
- Is the confidentiality of the information disclosed?
- Is the receiving agent authorized to get that corresponding information?

Definition of an agent

- “A software agent is a software with some inbuilt functionalities that interacts with other software agents and perform the allocated task based on the rules that govern them.”

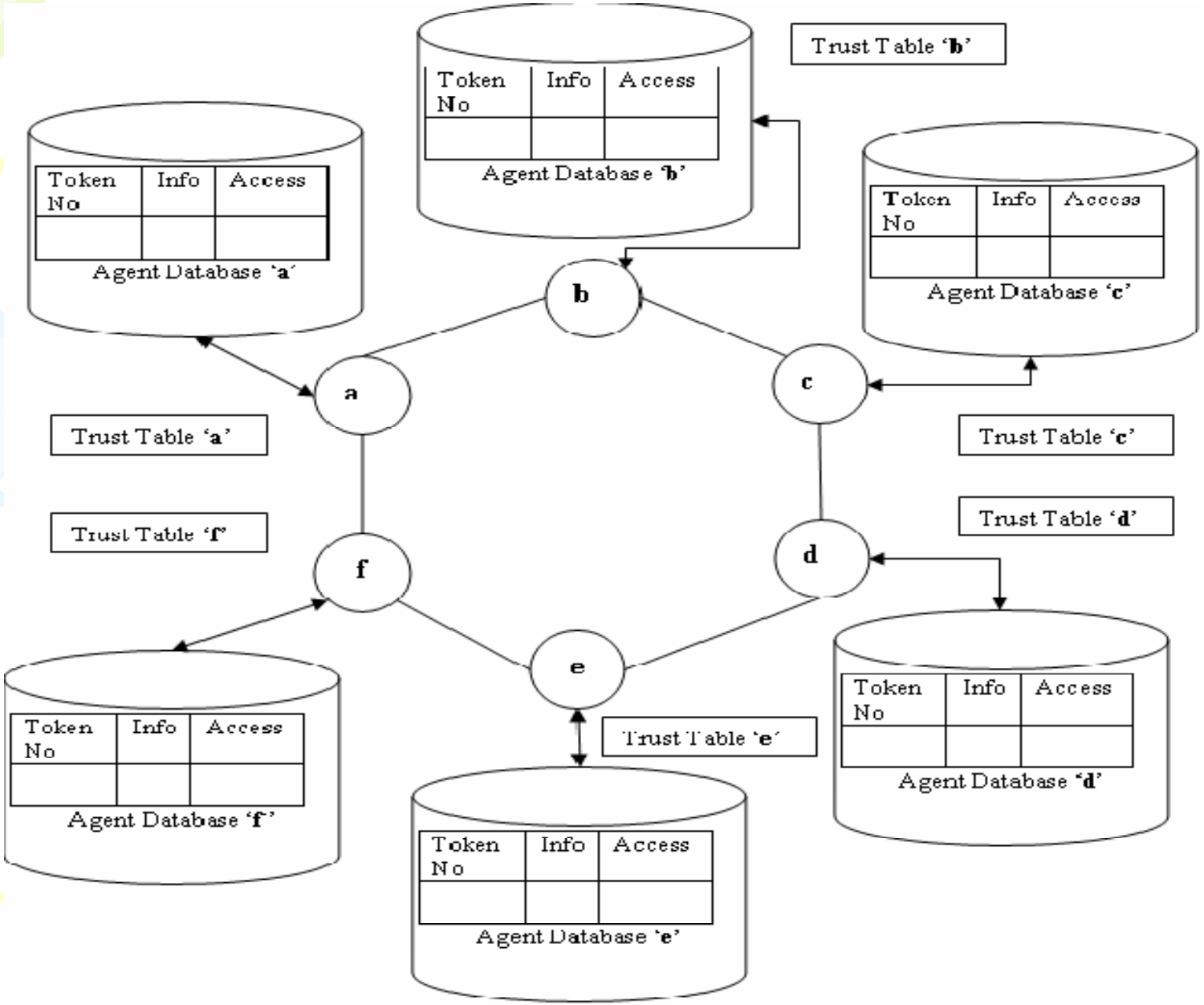
Functions of an agent

- Insight of changing environment
- Action required for the change
- Reason to the action taken
- Solution for the problem
- Draw Inferences and perform decision tree for future use.

System Architecture

- Multi Agent Network is similar to peer to peer communication.
- Each autonomous system in the network is an agent.
- Each agent knows its direct acquaintance alone. (Acquaintance – neighbor)
- All agents have their own set of information that they are going to share with the other agents

System Architecture



System Architecture

- In the following slides we will discuss about
- Confidentiality
- Trust Management

Confidentiality

- All the information have a confidentiality level linked to them.
- Agents can gain access to the information based on their level of trust.
- Four levels of Domains Denoted by D1, D2, D3 and D4
- D1< D2 < D3 < D4.
- Domains to maintain confidentiality. Confidentiality is linked with the information.
- Based on the trust level domain access changes.

Trust Management

- Trust in Agents are mainly based on the quality of the information they provide.
- There is also a trust table linked with each agent for all acquaintances.
- Trust levels have additive increase and Multiplicative decrease.
- There are minimal and maximum trust threshold.
- If the trust is below minimum threshold notifies other agents about corresponding bad agent and stops communication link.

Trust Management

- If the trust level is above threshold then more priority is given to that acquaintance.
- All requests from such acquaintance is serviced immediately.
- For any piece of information received the agents checks if it is available in its database. If it is available and there is some discrepancy with the content then trust is reduced.
- If same trust is increased. For all new information trust is increased.

Trust Management

- If same information arrives from two agents and content is different then content from agent which has higher trust level is accepted. The other agent's trust is reduced.
- Trust levels are maintained for future use. If new acquaintance minimum threshold trust is assigned.

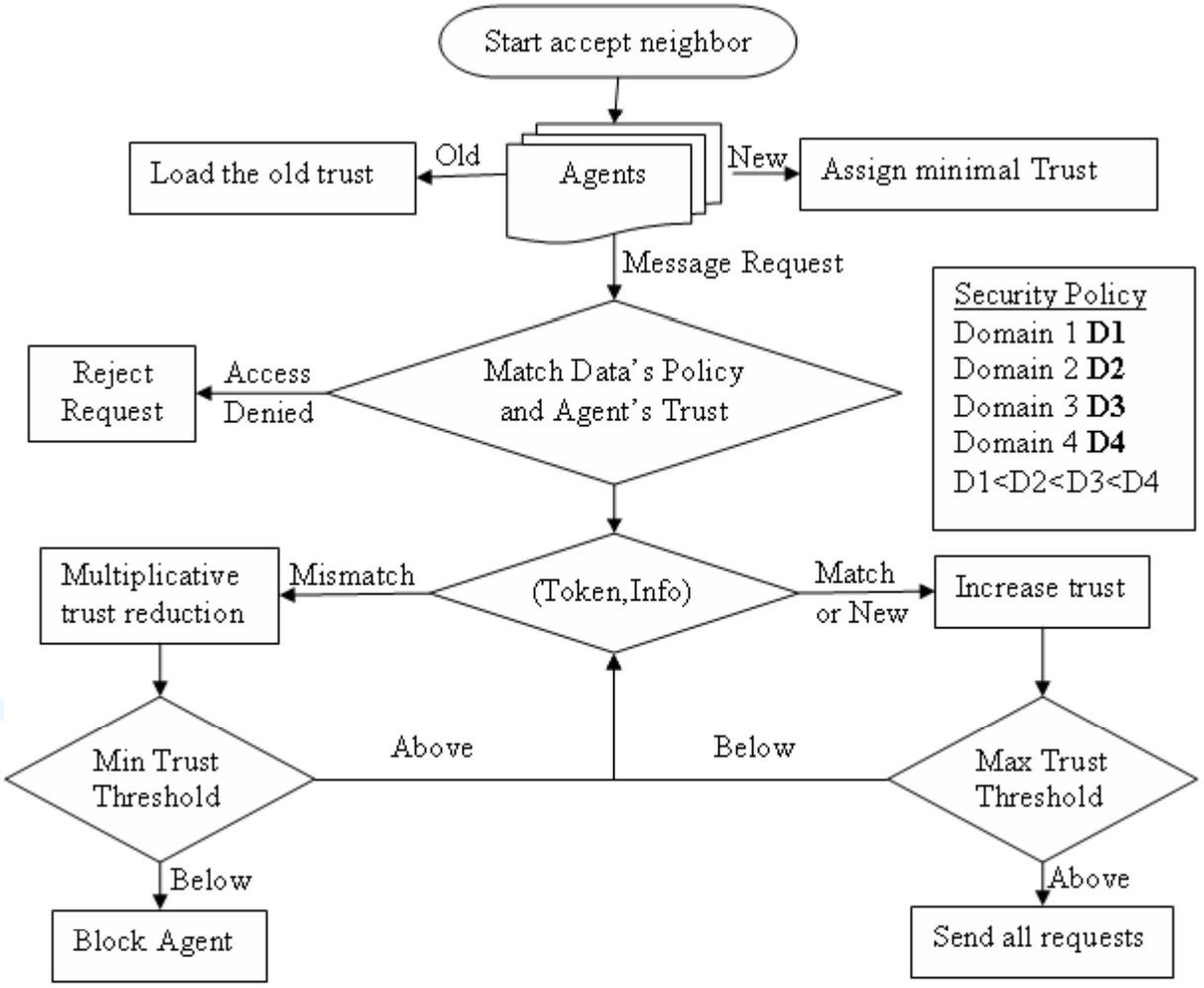
Algorithm for Information sharing within confidentiality and trust management

- Specifications of Algorithm.
  - Form Communication link with other agents where the neighbors are the acquaintances.
  - If new neighbor set minimal trust level else load the existing trust level from the database.
  - If an agent request for some information. Check the trust level of the agent and the access or security level linked with the information
  - If the access is granted allow service the request based on priority. Else reject request.
  - Start sending and receiving messages (the tokens and the Information are linked).

Algorithm for Information sharing within confidentiality and trust management

- If there is mismatch in messages multiplicative decrease of trust and if the trust goes below minimal trust after decreasing block agent and notify the network
- If there is message (new or old with match) additive increase trust and also if the trust is above max threshold send the entire request one by one.
- If any one agent has all information or end of session occurs end link store trust level, Message (Token and Information).
- Calculate the amount of data lost or gained from each acquaintance

Algorithm for Information sharing within confidentiality and trust management



Issues

- Entire update of the network model is not available any time.
- Agents do not know which agent requires the information.
- Scalability- (Size versus Performance Ratio is a very big issue).
- Network reliability.
- Relation between the information is very complex to find out.

Simulation

- Platform – Windows
- Programming Language – Java
- Programming Tool – Java Net Beans
- Peer- Peer Network Simulation .
- Each agent here in this simulation is a node.
- agents have security policy (Confidentiality of Info) linked with each information.

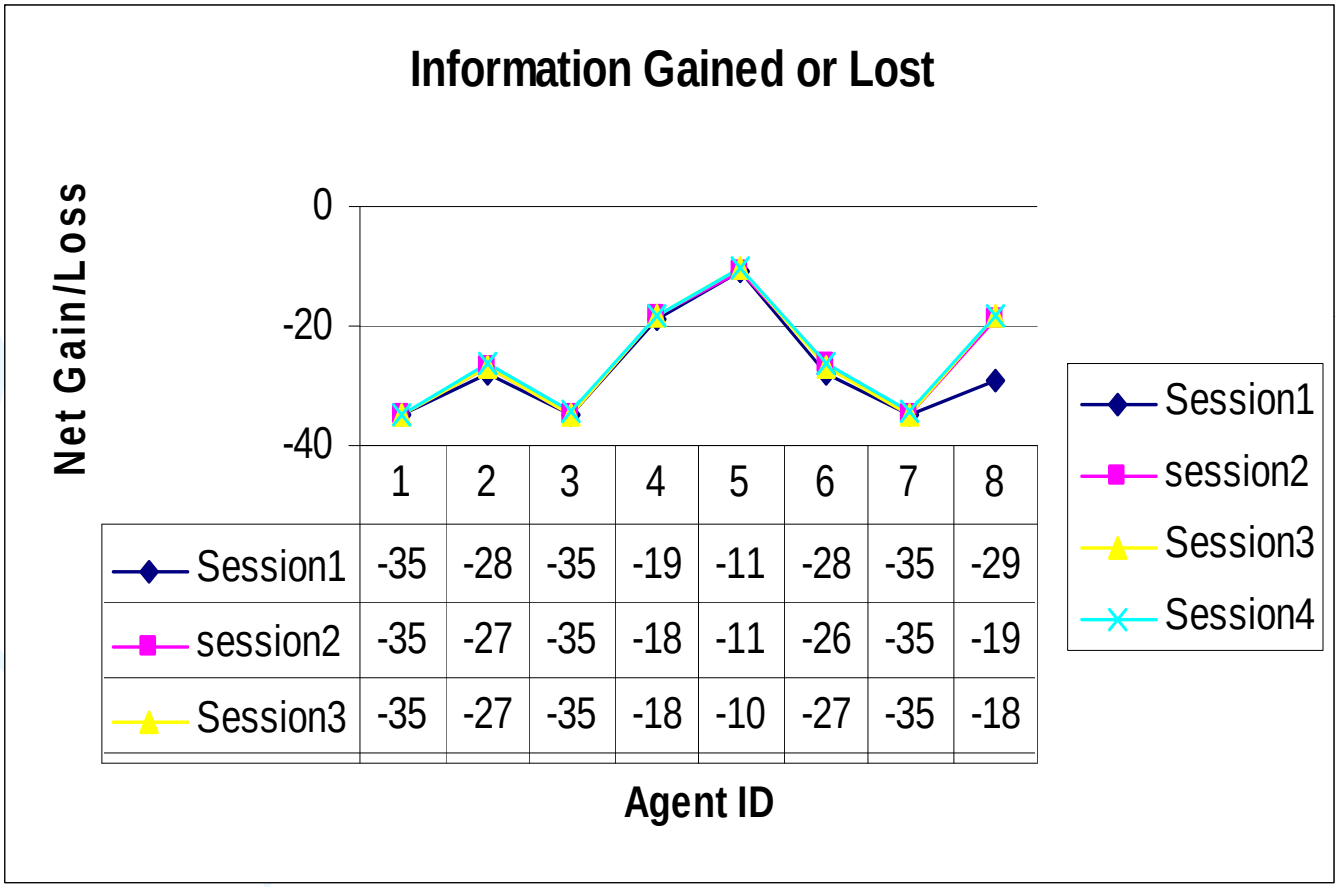
Simulation

- Agents maintain a trust level for the neighbors.
- There is also a token linked with all message.
- All agents start sending requests at different intervals mentioned in the configuration file.
- They collect the message based on the tokens. If an agent has token 5, It requests for 4 and 6 to gather some meaning from messages.
- The Simulation runs for 30 seconds or until some agent gathers all information required.

Net Gained/Lost Information

- Let T→ Net Gain/Loss of Information for any agent.
- R → The message received from Agents by some agent ai.
- S → The message sent to other Agents (a0, a1..... an) by agent ai.
- O → The own message of each agent in the beginning of the session.
- T = (R-(S+O))

Net gain of information in four continuous sessions of Simulation



Experimental Results

- The Chart has four set of simulations that was done within 8 agents.
- The simulation revealed that as the session proceeds the gain also increases.
- The reason for increase is that the agents begin to know their neighbors well.
- The trust level does not fluctuate that much and so the requests are serviced in an efficient way.

Future Directions

- Try and find out relationship between the different pieces of information.
- Negative message relation.

Conclusion

- We thank Dr. Robert Herklotz for funding our research on Information Operations Across Infospheres.
- We thank Profs. Latifur Khan,Murat Kantarcioglu, Elisa Bertino, Ravi Sandhu and Tim Finin as well as Dr. Mamoun Awad and Dr. Ebru Celikel
- We also thank the students Sai Chaitanya, Abinandhan Chandrasekaran, Ryan Layfield, Nathalie Tsybulnik, Li Liu, Alam Ashraful, Ganesh Subbiah, Gal Lavee, Kim Jungin and Pavan Chitumalla