Data Fusion Using Large Multi-Agent Networks: An Analysis of Network Structure and Performance

A. Knoll
University of Bielefeld, Faculty of Technology,
Department of Computer Science,
Postfach 100131, D-33501 Bielefeld, Germany

J. Meinkoehn
National Research Corporation for Mathematics and Data Processing (GMD),
Hardenbergplatz 2, D-10623 Berlin, Germany

Abstract – **The multi-agent paradigm may be used in many different ways for structuring the process of fusing data in large networks of information sources. The general goal of such networks is the maximisation of the quality of the result to be obtained at minimum cost, e.g. within a minimum time interval using many agents or before a given deadline using as few agents as possible. To this end agents compete for scarce resources or they work in parallel. Both the general granularity of the agent society and the competence assigned to each individual agent determine the information flow in the network. The multitude of parameters involved makes it difficult for the designer to optimally adapt the structure of the network to a given class of sensing tasks. In this paper we outline possible network structures and present an approach for analysing a number of important parameters characterising the network at a relatively abstract level. The abstraction enables a comparison of different network structures. The methods for the analysis may, however, be readily refined to evaluate a specific problem. A model of lateral coordination control in sensor networks is proposed as a result of this comparison. This model is based on the notion of negotiated cooperation between pairs of autonomous sensor agents. It is presupposed that the sensor agents may communicate with each other, thus enabling them to fuse the information they acquired. The actual cooperation phase is preceded by a bidding scheme to establish logical communication links. This scheme implements network self-organization, i.e. the automatic adaptation to changing environmental conditions. The cooperation is modelled on human social behaviour in the case of a task being worked on sequentially by team members with potentially different capabilities. A cooperation protocol is described to achieve the desired coordination. Our qualitative reasoning is supplemented by simulation results to support the hypothesis of the superiority of lateral over pure vertical coordination, particularly under severe environmental conditions, such as sensor failure.**

I. INTRODUCTION

As data fusion methods become more powerful and widespread, there is a natural tendency to design sensor systems with an ever increasing number of sensors contributing to the solution of a given sensing task. Each of the sensors is faced with the problem of making decisions based on its observation of a part of the environment and on partial a-priori information. The need for transferring information to locally disparate sensors and the need to associate their data both require a mechanism for transporting data of different structure

at minimal costs. To reduce the amount of data to be transferred, only those sensors that are necessary for the solution of a specific sensing task should be activated. This also makes it possible for the rest to work in parallel on the solution of other tasks. Consider a vision system with cameras of overlapping fields of view (e.g. for distributed vehicle monitoring [1]). The quality requirements of the task permitting, it is obviously desirable not to focus all cameras to a single specific object at one point in time, but to track different objects (possibly using the same sensor data). This is particularly important when the operations required to re-focus a sensor are costly.

It may also be very useful to fuse information on different aspects of one object using a set S_1 of sensors observing a certain spatial area A_1, while the information on a different area A_2 produced by the union of a subset of S_1 and a second set of sensors S_2 is processed by other fusion entities. This suggests another field of application of the multi-agent paradigm: The coordination of information processing entities, i.e. agents that do not necessarily comprise a physical sensor. There are three different interesting classes of the mapping of physical sensors to sensor agents:

- *The $1 \rightarrow M$ mapping:* One physical sensor provides information for M more or less specialised agents. In the field of Computer Vision the extreme view would be "one agent per pixel;" realistically, teams of agents are examined, which cooperate on the segmentation of regions [3]. Agent cooperation schemes must be highly flexible to adapt to varying scenes, and communication is a crucial issue.

- *The $1 \rightarrow 1$ mapping:* Sensors are equipped with local data (pre-)processing and communication facilities. The resulting homogeneous/heterogeneous network may in many respects be likened to distributed computer networks with a large number of nodes of equivalent/different capabilities.

- *The $M \rightarrow 1$ mapping:* This is the classical hierarchical network in which M sensors are controlled by one superior agent.

Clearly, a mix of the three is also possible, this would result in an $N \rightarrow M$ mapping, where N agents at a lower level interact with M agents at a higher level of a hierarchy. If a large sensor system is structured according to these schemes, the high number of nodes enforces a strategy for sensor coordination to achieve a common goal with minimal cost. This is the reason, therefore, that architectures must be developed to structure such sensor systems systematically, to organise them efficiently and to ensure a certain degree of fault tolerance by avoiding central controllers or coordinators (as was
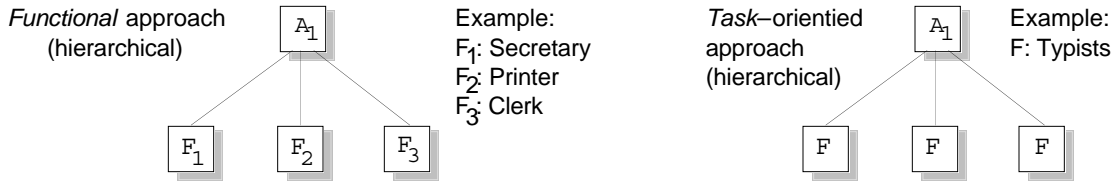
Fig. 1. Simple hierarchies. A_1: Task to be solved by the hierarchy. F_i: Specialised entities capable of solving A_1.
Left: At the lower level there are specialised agents.
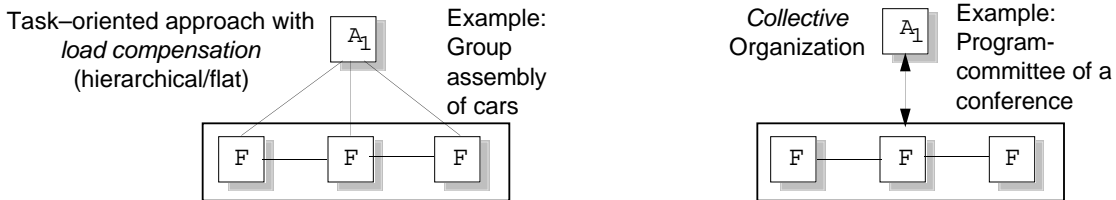*Right*: At the lower level complete solutions are obtained by non-specialised agents.

Fig. 2. *Left:* Extended hierarchy with load compensation. Non-specialised agents, which may be coordinated (identified) by the upper level agent.
Right: Lateral organization: At the lower level complete solutions are obtained autonomously by non-specialised agents. The latter
are not identifiable or at least not controllable by the upper agent.

demanded in [6]); see [5] for early work on the centralised approach. The latter two issues presuppose that strategies can be found to decide when to use a particular sensor for a set task. The relatively small amount of work that has been done in this field has focused on the vertical, i.e. hierarchical, structuring of coordination responsibilities within sensor networks. As common with all hierarchies, this approach offers a clear assignment of responsibilities as well as a mostly unidirectional and limited flow of control information. Unfortunately, its performance is impaired severely in the case of node failure. Moreover, with hierarchies it is difficult to add or remove nodes from the system without major system reconfiguration and they also suffer from the bottleneck syndrome, i.e. during high system load, tasks cannot spread out over the entire system although there may be resources available to process them. It will be shown below that a *flat* or *lateral* open (i.e. reconfigurable) structure offers significant advantages over hierarchical organizations as the complexity of the network increases, typically as its grows beyond sizes of teams of 15 sensor nodes.

II. Autonomous Agents

It is the purpose of large sensor networks to acquire information about the environment which is more comprehensive and more precise than the contribution from any single sensor. We assume that the network receives a sensing task from an external *mandator*. The entities of the network, the sensor agents then successively agree to form a group or *collective*. All members of this collective are capable of observing the same *object feature* (or complete object) and were assigned the competence to do so. This setting suggests a comparison with a human team of experts and it is therefore helpful to consider models from organization theory in order to cope with the problems of organization and communication within large networks. A huge amount of work has been done in this field, particularly for dealing with uncertainty introduced

when only partial information is accessible to every node (see, for example, [10]).

There are two main issues to be dealt with when organising teams of interacting agents [4]. The first of these issues is the *structure* of the team (sec. II.A) and the second issue is the definition of a *control mechanism* for coordinating the members of the team (sec. II.B). Criteria for selecting a structure and a control mechanism for a given network with a specific ensemble of sensor agents are both the *complexity* (e.g. the arrival rate of sensing tasks, the amount of knowledge necessary for resolving the problem and for coordinating the a priori knowledge and the resources) and the *uncertainty* (of acquired data, of the behaviour of the sensor agents and of the behaviour of the environment). The latter determines the number of agents necessary for completing the task. We shall now explore these issues further before proceeding to their application to networks of sensor agents.

A. Team Structure

A structure is specified by defining capabilities of the team members and by assigning responsibilities to them. In the case of sensor agent networks this implies that certain agents may specialise in particular tasks such as the sensing of the physical data; others work on different problems (e.g. preprocessing data from different channels, establishing communications paths or coordinating subordinate agents).

This differentiation of capabilities and responsibilities, however, is valid only for hierarchical structures, whereas in the case of the proposed lateral structure the agents are locally disparate but have equal rights and duties (as far as equal duties are possible for sensors of different physical principles). In a *simple hierarchy*, there exist a number of agents on a lower level, which are coordinated by an agent at an upper level. The agents at the lower level are all specialised to unique classes of tasks (fig. 1, left) or they may have universal capabilities (fig. 1, right). In either case they are subordinated in responsibility to the upper level agent.

In an *extended hierarchy* (such as proposed in [7]) there is more than one level of coordinating agents (fig. 2, left). Specialised agents may coexist with non-specialised agents in one network. If there are non-specialised agents at the same level, then there is a potential for these agents to coordinate themselves by interchanging information directly without any arbitration by a superior agent (fig. 2, right). This is, in essence, the concept of lateral structures. Both hierarchical and lateral structures may coexist in one network; subtrees are structured laterally and organise their cooperation within their layer of the subtree autonomously after receiving a certain task from their superior agent (or the external mandator). Finally, a network in which there are *only* lateral dependencies is called a *cooperative*. In such organizations there is no coordinating authority and agents may be members of different collectives that are working on different tasks. This structure of overlapping cooperatives forms the basis of our work because it also contains all hierarchical structures as a subset of possible specialisations (through the assignment of limited capabilities/competence to each individual agent).

B. Network Control Mechanism

The control mechanism defines *how* and *when* sensor agents communicate (interact). From an interaction, a transfer of control may result, which in turn is preceded by a selection process. The mechanism for coordinating communication between the agents may be either static, i.e. communication channels and hence groups of sensors for working on a certain task are fixed (e.g. [12]), or it may be dynamic. The latter means that cooperation between sensor agents is agreed upon for a limited period of time and vanishes after completion of the task. During the selection process, an exchange of information with different agents may take place and the decision for or against cooperating with a potential partner may be taken after evaluating the latter's offer in terms of promised result quality, e.g. time of completion and measurement precision.

An example of a static control mechanism is the *conservative* selection strategy: An agent which initiates a cooperation for a certain class of tasks for the first time looks for suitable partners and (possibly randomly) selects one of them. When the same task (or class of tasks) appears again, the agent selects the identical partner. After some time, all classes of tasks have caused each agent to "know" each partner for every task class and the partnerships for cooperation are fixed. With a dynamic strategy, current partnerships do not affect future relations. The selection process is repeated each time a cooperation becomes necessary and the momentary state of potential partners (e.g. workload of the potential partners) may be taken into account. Note that the selection strategy may have a drastic effect on the performance of the network. Moreover, the effects of sensor failure are less severe and the addition or removal of agents does not necessitate a complete re–initialisation of the network. Such dynamic strategies are obviously better suited for lateral networks in which agents are less specialised than in hierarchical networks where in certain situations there is only a limited choice of partners.

C. Lateral Networks of Autonomous Sensor-Agents

The general layout for the structure which forms the basis for the following discussion is shown in fig. 3. In principle, each sensor may become a member of any conceivable collective, i.e. a member of whichever collective promises the completion of a certain sensing task.
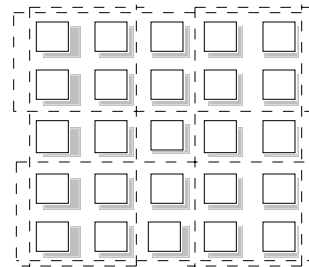

Fig. 3. Lateral sensor network. The dashed lines indicate four possible collectives each consisting of $K_S=10$ competent sensor agents. The centre agent is not included in either of these collectives. The size of the network is $K=5 \times 5$ agents. The external mandator is not shown; it may, however, communicate with each of the agents in both directions.

It is assumed that the network forms a grid of *K* nodes (i.e. sensor agents). A sensor agent is composed of a *coordination component* responsible for establishing links and transferring both control information and sensor data to other sensor agents together with an optional *sensing component* in charge of both data acquisition and data (pre-)processing. The sensing component and coordination component of a sensor agent communicate by means of a local data base, which stores information related to the tasks the sensor agent is allocated to.

A task comes into being once an object appears for the first time in the local field of view of any of the sensor agents (event). Since other agents may be necessary to resolve the task, the agent (or, originally, the mandator) may request that they contribute to the solution (demand). Moreover, there is normally an external demand from the mandator (e.g. from the robot control) to create certain results.

III. LATERAL AND DYNAMIC SENSOR AGENT COORDINATION

To establish contact between any two sensor agents of a collective and to coordinate their activities, a bidding scheme similar to the *Contract Net Protocol* introduced in [2] is utilised. This bidding scheme has been applied to the vertical distribution of subtasks in the context of automated manufacturing [11] and to the lateral scheduling of tasks in distributed systems [13]. In our context, the bidding scheme is applied to the lateral allocation of sensor agents for object identification and localisation tasks. Not only does this scheme enable the dynamic allocation of sensor agents within an individual cooperative, but it can also assign a sensor agent to different cooperatives thus coordinating the activities of overlapping cooperatives.

A. General Sensor Agent Interaction Patterns and the Contract Net Protocol

In an event-driven environment the agent starts processing the input when an object enters the local field of view. After generating preliminary results of a given quality, the agent proceeds to locate an interested external mandator by offering the preliminary results in a broadcast message. If interested (e.g. because of specific spatial task characteristics), the mandator responds with conditions constraining the amount of effort to be put into processing the task by the sensor agents. These conditions comprise two aspects which are important from the point of view of a mandator: the expected *quality* of the task processing results and the *time limit* within which re-

sults are expected. The time limit defines the time at which a mandator expects the results at the latest, whether they satisfy the quality constraints or not. In a demand-driven environment, the external mandator may also initiate a cooperation by itself, i.e. without being asked to issue a sensing task. After having received the constraining conditions of a mandator, the sensor agent continues to process the task:

- The generated preliminary results are immediately returned to the external agent if their quality already meets the external quality requirements.

- The sensor agent tries to allocate the complete task to another sensor agent of the collective corresponding to this task if it fails to meet the required deadline or quality.

The transfer of control between pairs of sensor agents is termed *contracting by negotiation*, the negotiation being the preceding bidding and selection phase which results in a transfer of control (in a contract). This contract assigns a certain credit to the sensor agent and commits it to perform a specific action. A sensor agent will refuse to bid for a task when it can derive its inadequateness from the task description and its own current status and capabilities (e.g. workload, state of sensory and processing resources, environmental conditions). If the time limit is exceeded, further processing of the task results must be carried out by the mandator or the application system because the generated results correspond to current sensor network capabilities and cannot be improved. A task ceases to exist in the sensor network when its results are communicated to a mandator.

Contracting by negotiation enables a sensor network to flexibly adapt to indeterminate environmental and agent specific conditions governing its performance. It also leads to the sequential coordination of only the minimum amount of resources required to solve a task according to given quality constraints. The sensor agents not participating in the processing of a task of a particular collective remain free to employ their resources in other collectives. Due to the lateral relations between sensor agents, sensory results from multiple, possibly disparate sources can be accumulated and integrated. Furthermore, such a sensor network is fault–tolerant and extendible because new agents or enhanced capabilities of already existing agents are included automatically in the contracting scheme. The partial or complete failure of an agent is tolerated; a totally disabled agent will automatically not be considered in the coordination of the remaining network.

A great advantage of the lateral organization is the immediate availability of all other agents (i.e. the maximum network capacity) to each sensor agent if need be. This is in contrast to a hierarchical structure where contact between nodes at the same hierarchical level is possible only through intermediate managing nodes at higher hierarchical levels. In the vertical case, a managing agent coordinates only a subset of network capacity for a specific set of tasks. Coordination

is efficient only in such a subset because otherwise too many intermediate nodes may be involved, thus inhibiting tasks from spreading out over the entire network. The relevance of this property increases as complexity and uncertainty in the network environment grow.

B. The Cooperation Protocol

A common *frame of reference* (or "common language") is a precondition for cooperative object identification and localisation by multiple agents. This may be given by a global coordinate system referencing fixed points in Euclidean space. Additionally, a globally utilised description of agents, their capabilities and of the tasks to be processed must be provided. An *object description* comprises a unique object type identifier and parameters determining its current position, alignment and size (fig. 4). In addition, an object description includes *quality factors* as a measure of reliability of the slot values describing an individual object. A quality factor is normalised to the interval [0,1], with a quality factor of 1 expressing complete reliability of the associated object slot value. A quality factor is the result of a specific sensor data evaluation. A *sensor agent description* encompasses information about the basic capabilities of a sensor agent and its current status, as well as administrative information. The status of a sensor agent is described by values expressing its current workload and sensory precision/variance. The workload is given as the number of tasks the sensor has successfully bidden for but not yet processed. Due to the generally sporadic time of arrival of individual tasks and the indeterminate amount of time required for their processing, this is the only way of pragmatically measuring the workload of a sensor agent. A *task description* is composed of administrative information, the conditions constraining the cooperative processing of the task as given by a mandator and results generated by sensor agents which have already processed that task. The administrative information consists of a unique task identifier and the communication address of the mandator, both supplied by the agent. The task description also includes a unique identifier of the local field of view of the sensor agent where the task originated from. The external constraints consist of a value defining the task processing time limit and the desired quality factors for the object identification and localisation relevant slots of the object description. To facilitate contracting by negotiation among sensor agents, appropriate message types must be defined. Five message types are used:

- A *request for bids*-message describing a task to be processed cooperatively. It initiates a negotiation and selection phase.
- A *bid*-message by which an interested sensor agent offers its capacity to process a task.
- An *award*-message by which an initiating sensor agent transfers task information to the selected bidder.

```
type object is                          type sensorAgent is                      type task is
   objectID          : integer;            address         : string;               taskID            : integer;
   position          : vector;             localRangeID    : integer;              mandatorAddress   : string;
   alignment         : vector;             qf_objectID     : real[0,1];            localRangeID      : integer;
   size              : real;               qf_position     : real[0,1];            timeLimit         : real;
   qf_objectID       : real[0,1];          qf_alignment    : real[0,1];            qf_objectID       : real[0,1];
   qf_position       : real[0,1];          qf_size         : real[0,1];            qf_position       : real[0,1];
   qf_alignment      : real[0,1];          workload        : integer,              qf_alignment      : real[0,1];
   qf_size           : real[0,1];          precision       : real;                 qf_size           : real[0,1];
end object;                             end sensorAgent;                           results           : list of object;
                                                                                end task;
```

Fig. 4. *Left*: Object description, *Centre*: Sensor agent description, *Right*: Task description

- A *request for interest*-message by which a sensor agent offers to mandators further processing of a newly arrived task.
- A *result*-message by which a sensor agent currently allocated to a given task returns the available results to a mandator when either the quality requirements for this task have been met or its time limit has expired.
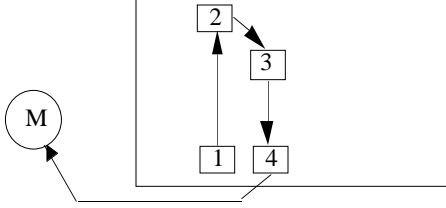

Fig. 5. Multiple agents form a team to process a single specific task. Agent 1 was awarded the task originally, but turned out not to be capable of meeting the requirements. Note that all but the active agent may simultaneously be members of others teams working on different tasks.

Fig. 5 shows how a collective of $K_S=4$ agents is assembled if the first agent that was awarded the task turns out to have been too optimistic, i.e. it cannot meet the requirements. It is important to note that, although preferences for choosing particular collectives may exist, the structure of the team (i.e. members of the collective) is not set a priori and cannot be predicted. It is only after the completion of the task that the individual members of the team can be identified.

IV. EVALUATING THE PERFORMANCE OF ORGANIZATION SCHEMES

We now turn to the interesting question of how well the architecture performs under several conditions.
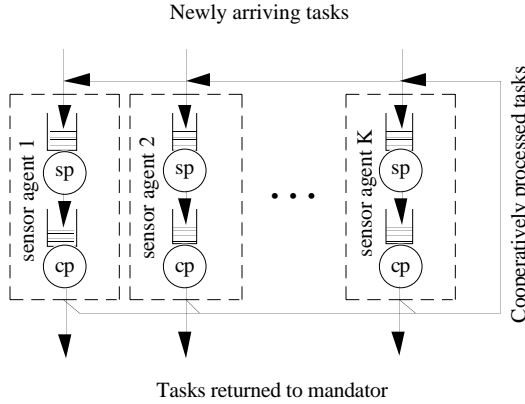

Fig. 6. Sensor agents as a queuing network

The performance of the agent organization is assessed by steady-state simulation, which models a sensor network as a set of interconnected service centres equipped with queuing facilities (fig. 6). Each agent is modelled as consisting of two sequentially related service centres, i.e. its sensor component (*sp*) and its coordination component (*cp*). Agents without a physical sensor (coordinating agents) lack the sensor component. Among the network service centres, transactions circulate which represent object identification and localisation tasks. For the sake of simplicity, only sensor agents with homogeneous capabilities and tasks of similar complexity are considered.

A. Simulation Model

The sensor component of a sensor agent is represented as an M/M/1-queue, i.e. a service centre with exponentially distributed inter–arrival times of new tasks and exponentially distributed service time. The coordination component is represented as an M/D/1-queue, i.e. with exponentially distributed inter-arrival times and constant service time.
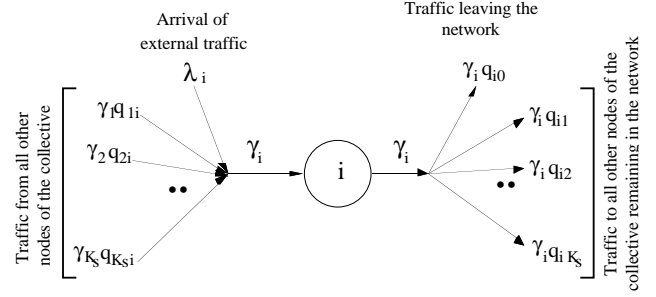

Fig. 7. Task traffic at sensor agent i

For the purposes of our simulation, a task arriving at an agent is first processed by its sensor component and then by its coordination component. In particular, it is assumed that an external mandator was already located for each newly arriving task. The rate of newly arriving tasks (e.g. due to object movement) λ_i at a sensor agent i with $i = 1,...,K$ is termed *external arrival rate* and is assumed to be identical for all agents. Thus, the total external arrival rate is given by $\lambda = \lambda_i K$. A task processed by a sensor agent i is routed to a sensor agent j of the corresponding collective of K_S agents (which are competent to work on the task) with probability q_{ij} where $i, j = 1, …, K_S$. The task exits the network when it was successfully completed with probability

$$q_{i0} = 1 - \sum_{j=1}^{K_s} q_{ij} \text{ with } i = 1, …, K$$

The probabilities q_{ij} are called the *network routing probabilities* [9]. The tasks arriving at agent i from other agents j (because of contracting) are a fraction of the total rate of tasks γ_j leaving sensor agent j with $j = 1, …, K_S$. The rate of traffic flowing into agent i is called the *internal arrival rate* of agent i and is given by

$$\sum_{j=1}^{K_s} \gamma_j q_{ji} \text{ with } i = 1, …, K$$

Due to the flow balance assumption [8], tasks must *leave* a sensor agent at the same rate at which they *arrive* there. A fraction q_{ij} of the set of tasks arriving at agent i are directed from sensor agent i to sensor agent j with the rate $\gamma_i q_{ij}$. Furthermore, a fraction q_{ji} of tasks is directed from sensor agent j to sensor agent i with the rate $\gamma_j q_{ji}$. Consequently, the total traffic rate γ_i at a sensor agent i is given by the *network traffic equations* (see fig. 7):

$$\gamma_i = \lambda_i + \sum_{j=1}^{K_s} \gamma_j q_{ji} \quad i = 1,...,K$$

The external arrival of tasks is considered a stationary Poisson process. However, the internal arrival rate is not necessarily such a process: in the case of a dynamic selection

strategy (such as selection by smallest workload) arrival rates depend on system history. Moreover, as will be explained below, the probability q_{ji} of a task arriving from sensor agent j at sensor agent i is a function of the number of sensor agents which have already processed this task. Therefore, our performance evaluation is conducted by means of simulation instead of analysis because the non-Poisson characteristics of the processes significantly complicates an analytical approach.

The coordination component of an agent decides by means of an evaluation function whether a task processed by the sensor component can be successfully completed. As no further assumptions on the nature of sensor data evaluation were made, the process of K_S agents transferring a given task and accepting it for completion or rejecting it, is viewed as a Bernoulli experiment. After each transfer the task is accepted by the new agent with probability b and rejected with probability $1 - b$. The probability of the kth agent accepting the task for completion is then given by

$$P(k) = (1 - b)^{k-1} b$$

The corresponding geometric probability distribution is given by

$$F(k) = 1 - (1 - b)^k$$

This function determines the probability q_{i0} of a task exiting the network as successfully completed by sensor agent i after passing k agents including i ($k \geq 1$) with $E[k] = 1/b$. Additionally, q_{i0} is set to 1, should the set processing deadline have expired at the time a task arrives. Based on these assumptions, vertical and lateral structures were compared in performance.
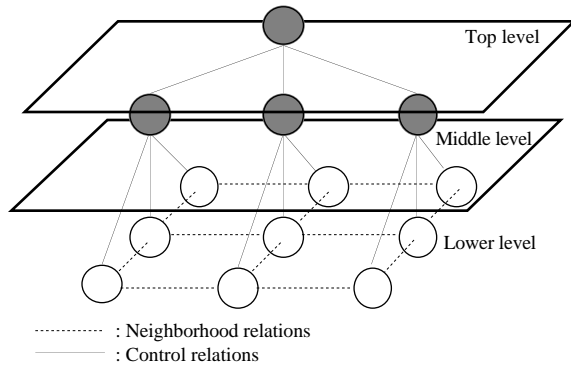

Fig. 8. Sensor network hierarchy

For the vertical organization, two additional layers of sensor agents (manager–agents) were introduced with the original sensor grid constituting the lowest level. Each sensor agent at the middle level coordinates exactly one row of the sensor agent grid. The middle level agents are coordinated by a single manager–agent at the top level (fig. 8). At the middle and top level a task is processed only by the coordination component of a manager–agent. Specifically, a sensor agent at the middle level coordinates only a subset of the collectives defined by its subordinate sensor agents.

The main simulation output parameter of interest and hence the measure of organization performance used for comparison is the percentage V of tasks successfully completed within a given deadline. The following variables were among the sim-

ulation input parameters, which were introduced to determine the behaviour of the modelled organization and, consequently, the value of V: The *network size K* defining the number of sensor agents in the network; a relative *processing deadline d* within which tasks should be completed; a *failure probability f* which determines whether a sensor agent fails at a specific point in time; a *repair delay r* after which failed sensor agents return into the system; a completion *probability b* determining the mean number of sensor agents required to successfully complete a task

B. Simulation Results

The selection strategy employed in both a lateral and a vertical organization is selection by smallest workload. As a measure of difficulty of the task, the coefficient b was varied, a decrease in b resulting in an increase in E[k], the mean number of sensor agents necessary to successfully complete a task. The node failure probability f and repair delay r as well as the network size K are viewed as measures of complexity. Additionally, the coordination component service time cp was varied to represent an increasing complexity in reaching coordination decisions as opposed to the sensor component service time sp.

Figs. I…II depict the effect of increasing the network size K while the other parameters remain fixed, except for the failure probability f, which was 0.01 and 0.05, respectively. In addition, organization performance (fl denotes a lateral and hi a hierarchical organization) is shown for different degrees of task uncertainty b. In the case of low failure probability f (fig. I), the superiority of lateral over vertical organization is evident because even with small network sizes the lateral organization provides a higher percentage V of tasks completed successfully within the deadline d. This advantage increases with growing network size K and uncertainty b. Particularly, it is shown that for a large network (K = 49) an increase in uncertainty leads to significantly less degraded performance when compared to the vertical organization. In this case, with b decreased from 0.5 to 0.33 (and hence E[k] increased from 2 to 3) the lateral organization suffers from a performance degradation of ca. 5% (as given by the parameter V), whereas the vertical organization performance degrades by approximately 20% under identical conditions. A further increase in complexity (high failure probability $f = 0.05$; fig. II) yields an important result: initially, i.e. with small network sizes, the vertical organization exhibits a better performance than the lateral organization. However, as K increases, a break–even point is reached, at which the lateral organization performance exceeds that of the vertical organization. Moreover, as uncertainty increases, this break–even point occurs at decreasing network sizes. At first sight, this fact may look contradictory to our argumentation; note, however, that the difference in performance in the two organization types grows with increasing difficulty as the network size increases.

The results displayed in figs. I…II are supported by figs. III…IV, which show the organization performance as a function of the repair delay r with high failure probability f fixed at 0.05. The repair delay corresponds, for example, to the time it takes to re-focus a sensor if the current focus turns out to be inadequate. The probability f indicates how frequently this happens. Initially, with a small network ($K = 9$) and short repair delay, lateral organization is at an advantage over vertical organization. Soon, however, with increasing repair delay, lateral organization performance degrades significantly below vertical organization performance (fig. III). This situa-

tion is completely different in the case of a large network ($K=$ 49; fig. IV): Here, even with very long repair delays r, the lateral organization significantly outperforms the vertical organization. It is also clearly visible that the difference in organization performance increases with growing uncertainty. This sensitivity of the vertical organization to increased network size is explained by the bottleneck effect affecting managing agents.

A different measure of complexity is the amount of time required by a coordination component to reach a coordination decision. It was considered for varying circumstances and the corresponding results are displayed in fig. V. The network size has no significant effect on lateral organization performance, but very much on vertical organization performance. Here, another interesting feature of vertical organization performance was encountered: As cp is increased (and sp correspondingly decreased), lateral organization performance remains relatively stable, rising from nearly 100% to a full 100% of successfully completed tasks. This is largely due to the growing influence of the constant service time cp and, correspondingly, the diminishing influence of the exponentially distributed service time sp. This is a relevant setting for networks that consist of a large number of coordinators (that do not have a sensing component). However, besides being sensitive to increased network size due to bottleneck potential, vertical organization performance rises sharply with increased service time cp. It reaches an optimum in the vicinity of the lateral organization performance, and declines just about as sharply as it has risen before reaching the optimum. The results displayed in fig. V suggest that, in contrast to the robustness of lateral organization performance, a vertical organization is highly sensitive to the relation of sp to cp service times. Thus, a vertical organization is only justifiable if this relation results in optimal or near–optimal performance. It seems to be increasingly difficult to establish the according range of service time values guaranteeing such performance with increasing network sizes. The right–hand sides of the performance plots of the vertical organization are again explained by the bottleneck characteristic of managing agents which is directly amplified by increasing cp. The left hand–sides, however, are not so easy to explain: With decreasing cp service times, tasks may flow increasingly faster through the hierarchy, leading to saturation effects in the collective subsets coordinated by the middle level managers. This presumption is supported by fig. VI, which displays the mean task population of hierarchical organizations with cp service times corresponding to those shown in fig. V. This saturation effect within the collective subsets diminishes with increasing service time cp until the performance reaches the optimum, and is afterwards converted to the bottleneck effect mentioned above. The performance increase occurring with increasing service time cp is explained by the fact that the traffic originally (with very low cp) leading to neighbourhood saturation is increasingly delayed at the corresponding middle level manager. This increase in delay has an advantageous effect on vertical organization performance until it reaches the point where the bottleneck effect induced by that delay outweighs its advantageous effect.

V. Conclusions

The subject of sensor coordination and control may currently appear esoteric, but it will soon turn out to be quite relevant as the complexity of these systems continues to increase and the prevailing ad hoc approaches will no longer provide ade-

quate solutions. This does not imply, however, that systems with a smaller number of sensors, such as with current mobile robots, could not profit from a well–structured organization and coordination of their sensor subsystem. This aspect comes into play when a smaller number of sensors is employed in a constantly changing scenery. In order to acquire a maximum of environmental information, it is obviously desirable to supply the sensor network with a large number of goal–directed sensing tasks. The coordination protocol must then ensure a high degree of "liveness", i.e. the tasks must be worked on in parallel by as many sensor agents as possible.

It was shown that lateral control in distributed sensor networks is feasible through a corresponding cooperation protocol motivated by considering models from organization theory. Furthermore, simulation studies have revealed not only a general quantitative superiority of lateral over pure vertical control structures, but also an increased sensitivity of vertical organizations to growing complexity and uncertainty when compared to lateral organizations. However, a clear performance–bound distinction between lateral and vertical organization will not be possible without detailed and comprehensive experimenting by simulation *and* real multi–sensor systems. In fact, the simulation results presented here indicate that issues of complexity and uncertainty are closely coupled and can not be studied in isolation.

VI. References

[1] Carver, N., Lesser, V., "Distributed sensor interpretation: Modeling agent interpretations in DRESUN", Comp. Sc. Tech. Report 93-75, University of Massachusetts, Amherst, 1993

[2] Davis, R., Smith, R., "Negotiation as a metaphor for distributed problem solving", *Artificial Intelligence 20 (1)*, 1983

[3] Demazeau, Y., "From feature extraction to integration of visual modules using agent systems", *Proc. Workshop "Solving Complex Problems with Multi-Agent Systems"*, Univ. of Bielefeld, Feb. 1994

[4] Fox, M., "An organizational view of distributed systems", *IEEE Trans. on Systems, Man and Cybernetics, SMC-11*, 1981

[5] Henderson, T., Fai, W., Hansen, C., "MKS: A multisensor kernel system", *IEEE Trans. on Systems, Man and Cybernetics,* SMC-14, 1984

[6] Iyengar, S., Kashyap, R., Madan, R. Thomas, D., "A tree–structured sensor fusion architecture for distributed sensor networks", *SPIE Vol. 1306 Sensor Fusion III*, 1990

[7] Iyengar, S., Sharma, M., Kashyap, R., "Information routing and reliability issues in distributed sensor networks", *IEEE Trans. on Signal Processing*, Vol. 40, No. 12, Dec. 1992

[8] Lazowska, E., Zahorjan, J., Graham, G., Sevcik, K., "Quantitative system performance – computer system analysis Using queueing network models", Prentice-Hall, 1984

[9] Mitrani, I., "Modeling of computer and communication Systems", Cambridge Computer Science Texts 24, Cambridge University Press, 1987

[10] Pete, A., Pattipati, K., Kleinman, D., "Distributed detection in teams with partial information: A normative descriptive model" *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 23, No.6, Dec. 1993

[11] Parunak, H., "Manufacturing experience with the contract net", in: Michael Huhns (Ed.), "Distributed Artificial Intelligence", Pitman, 1989

[12] Rao, B., Durrant-Whyte, H., Sheen, J., "A fully decentralized multi-sensor system for tracking and surveillance", *Int. J. Rob. Res.*, Vol. 12, No. 1, Feb. 1993

[13] Ramamritham, K., Stankovic, J., Zhao, W., "Distributed scheduling of tasks with deadlines and resource requirements", *IEEE Transactions on Computers 38 (8)*, 1989
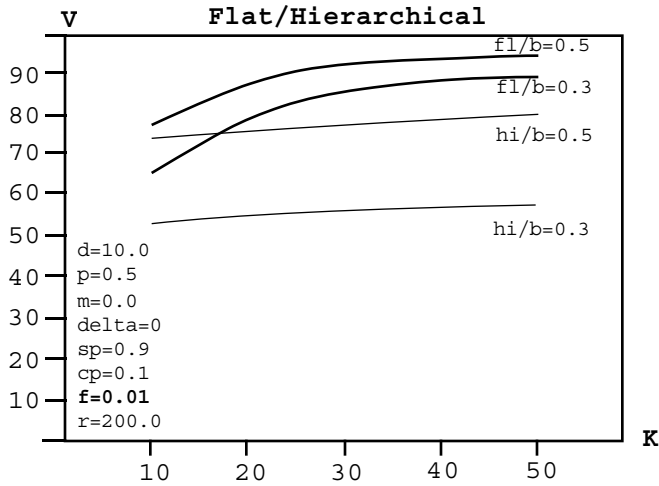
Fig. I. Effect of network size K on V with varying uncertainty (completion probability) b. Low failure probability f and long repair delay r.
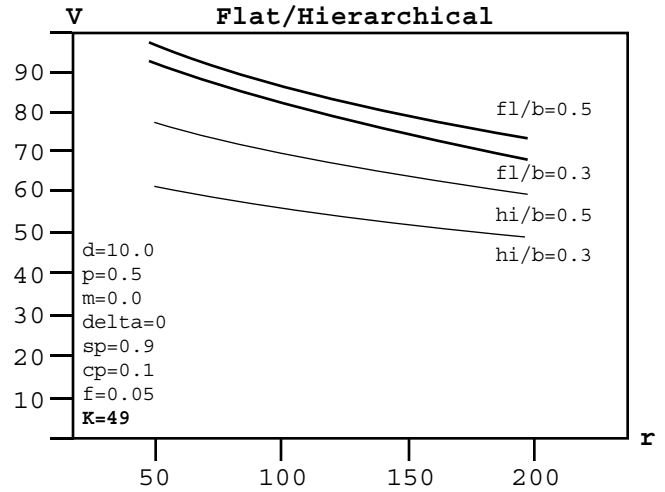

Fig. IV. Effect of repair delay r on V with varying uncertainty (completion probability) b. High failure probability f and large network size K.
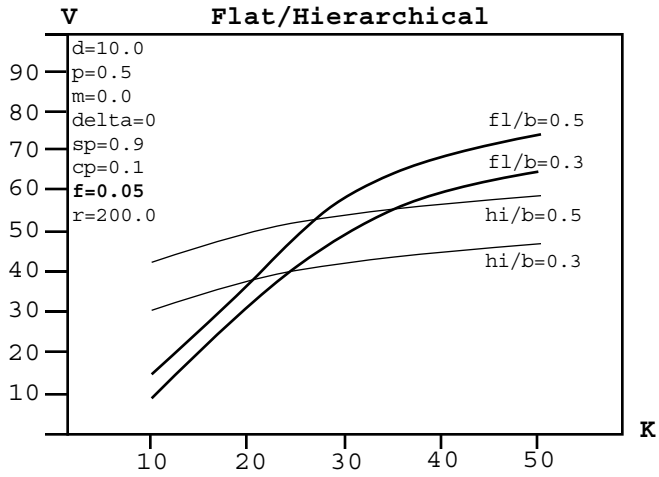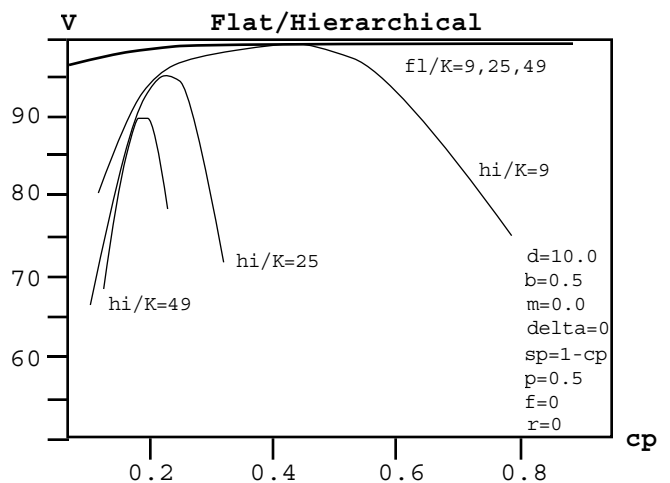

Fig. II. Effect of network size K on V with varying uncertainty (completion probability) b. High failure probability f and long repair delay r.

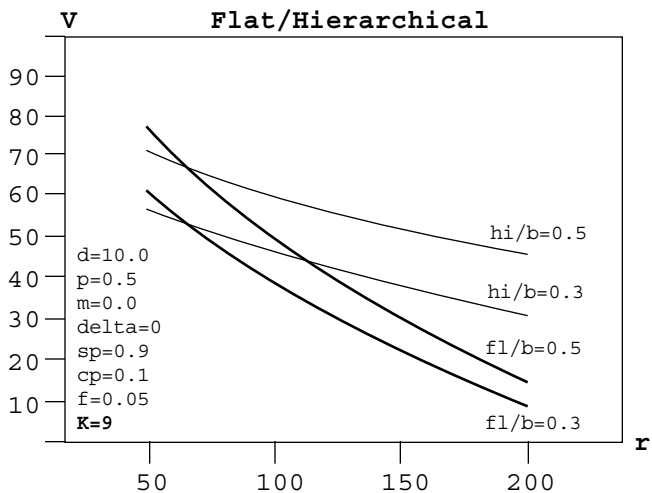Fig. V. Effect of coordination component service time cp on V

Fig. III. Effect of repair delay r ("re-focus delay") on V with varying uncertainty (completion probability) b. High failure probability f and small network size K.
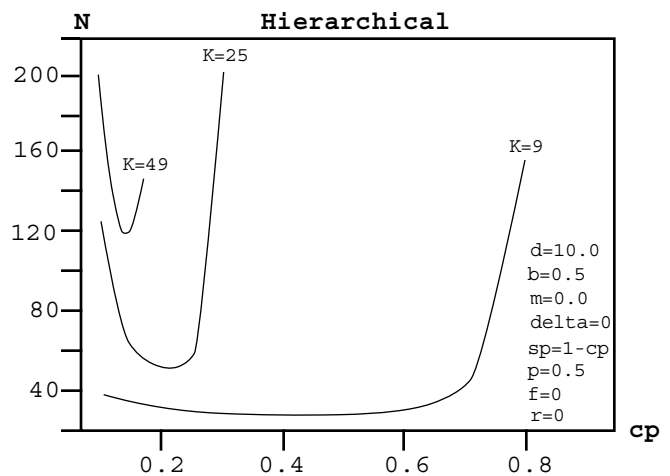

Fig. VI. Effect of coordination component service time cp on population N (=tasks waiting to be serviced)