A General Computational Framework for Distributed Sensing and Fault-Tolerant Sensor Integration

Lakshman Prasad and S.S. Iyengar

Dept. of Computer Science

Louisiana state University

Baton Rouge, LA - 70803.

Abstract

The design of Distributed Sensor Networks has to take into consideration sensor failures that occur while functioning in the real (physical) world. This demands a technique of integration of sensor information that is fault-tolerant so that the network is reliable for target recognition and tracking problems.

In our earlier paper ~~[LIKM] we~~ proposed a computational characterization of fault-tolerant integration of abstract sensors that were 1-interval estimates.

In this paper, we propose an abstract framework to address the general problem of fault-tolerant integration of sensor information in a general distributed sensor network. The essential ideas of this abstract framework stem from certain rudimentary notions in the theory of differentiable manifolds. This framework addresses a very general distributed sensor network both at the local level of sensor data integration at distributed processors as well as global exchange and assimilation of information available at various processors in the network. This paper is a continuation of our earlier work [LIKM].

Keywords & Phrases: Fault-tolerant integration, distributed sensor network ,charts, local coordinates, transition functions, sensor-clusters, parameter space.

* This author's work was supported in part by the Office of Naval Research under Grant No. ONR-N00014-91-J-1306 and in part by the LEQFS- Board of Regents under Grant No. LEQFS-RD-A-04.

TABLE OF CONTENTS

1.0 INTRODUCTION

The distributed sensor processing problem, in the context of distributed sensor networks, involves both detecting and tracking of multiple moving objects has been shown to have a wide ranging applications in areas such as particle physics, medical imaging, radar tracking etc. One of the features that distinguishes distributed sensor processing is its demand for the development of a computational framework for sensor integration. This includes the problem of fault-tolerant integration of information from multiple sensors, mapping and modeling the environment space and task level complexity issues of the computational model. Further these techniques have to be robust in the sense that even if some of the sensors are faulty, the integrated output should still be reliable. For details on multi-sensor integration and fusion in intelligent systems, see([Hube 81], [KaOM 90], [BlBr 90], [LuKa 89], [HAMD 87], [LuLS 88], [Duwh 88], [Zhen 89] and [GeCh 90]). In this paper, an abstract paradigm for distributed sensing and fault-tolerant sensor integration is discussed.

1.1 Scope of this paper

This paper's main objective is to propose a new computational model for distributed sensor network problem at the local level of sensor data integration at distributed processors as well as global communication and assimilation of information available at various processors in the network.

The distinguishing feature of our computational paradigm over the previous model[LIKM] is in addressing a broad-based computational framework which can accommodate a wide range of sensors and a variety of fault-tolerant integration techniques depending upon the phenomenon being sensed and the method of sensing

The central idea of our abstract framework stem from certain rudimentary notions in the theory of differentiable manifolds.

The global picture of the integration phenomenon is addressed using local coordinates of the environment space and the corresponding transition functions.

1.2 Organization of the Paper

In section 2, we describe earlier work related to this paper. In section 3 an abstract model for general distributed sensor networks is proposed. In section 4 a scheme for comparing and patching together the local information about the behaviour of the parameter being measured, in order to obtain a global picture of the parameter's variation is put forward.

2.0 RELATED WORK

Marzullo [Marz 89] considers the case of a processor receiving input from several sensors whose outputs are connected intervals. He gives a fault-tolerant integration algorithm which takes as input the intervals representing the sensors and gives as output of the processor a connected interval representing the sensor values. More precisely: Let there be n sensors, each of which yields an interval as its output. these sensors measure a certain physical value and their intervals contain the physical value unless they happen to be faulty sensors.

Thus, a correct sensor is one which contains the actual physical value in its interval. Any two correct sensors must overlap since they both contain the physical value being measured.

Marzullo considers the case when atmost f sensors are faulty and gives an algorithm which yields a connected interval as the output of the processor, containing the physical value.

If atmost f of the n sensors are faulty, then it follows that at least n-f sensors are correct. Marzullo considers all possible nonempty (n-f) intersections of the n-sensors. A sensor which does not belong to any of the (n-f)-cliques is faulty since a correct sensor overlaps with at least (n-f-1) other correct sensors. One and only one of the (n-f)-intersections contains the physical value. Since it is not possible to decide which intersection has the physical value (which is as yet unknown to us) and since the processor output is required to be a connected interval, the smallest connected interval containing all the (n-f)-intersections is taken to be the output of the processor. It is easy to see that it contains the actual physical value. The wider this interval is, the lesser the accuracy of the processor output. Marzullo proves the existence of bounds for the width of this interval in terms of f.

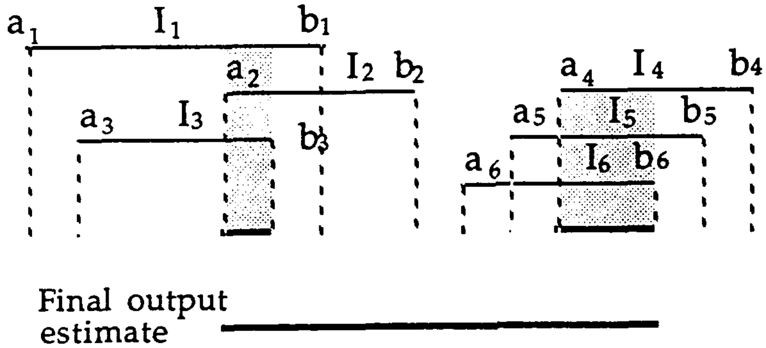

Figure 1. Integration of Interval estimates

$(a_1 < a_3 < a_2 < b_3 < b_1 < b_2 < a_6 < a_5 < a_4 < b_6 < b_5 < b_4)$

In our earlier work[LIKM 91], we provide a functional characterization of fault-tolerant integration of 1-interval estimates considered by Marzullo and narrow the width of the output estimate in the case when the sensor faults are "tame", However, the technique of integration and its functional characterization did not in anyway depend upon the fact that the sensor outputs considered have intervals on the real line. Indeed, we may look at distributed sensor networks and fault-tolerant sensor integration from a more general point of view, as we shall do shortly.

A distributed sensor network consists of several clusters of sensors distributed in space. Each cluster of sensors reads the value of a parameter in a certain region. All the sensors in a cluster read the same value. This redundancy is required to ensure fault-tolerance since some of the sensors in a cluster may be faulty. A fault-tolerant technique of integration is to be employed to obtain a reliable estimate of the measured parameter's value from the outputs of the sensors in a cluster. The scope of all the sensors in a distributed sensor network taken together is the region under observation. The parameter being measured may be in general a vector(e.g., velocity, position, etc.) Each sensor typically gives a set of values(e.g., a connected interval or a connected region) as its output estimate of the parameter instead of a single value, this being due to the uncertainty of the sensor.

Keeping the above remarks in mind, we present an abstract paradigm for distributed sensing and fault-tolerant sensor integration in the next section.

3.0 ABSTRACT REPRESENTATION OF A DISTRIBUTED SENSOR NETWORK

3.1 Notation and Definitions

We Formally introduce here the abstract setup for a general distributed sensor network:

Definition 1: An abstract distributed sensor network is a 4-tuple ($\mathcal{X}$, $\left\{\mathcal{E}_i\right\}_{i=1}^m$, $\left\{\mathcal{S}_i\right\}_{i=1}^m$, $\mathcal{P}$), where:

i) $\mathcal{X}$ is the space under observation by the sensors called the *environment space.*

ii) $\mathcal{P}$ is the space of all possible values of the parameter being measured. If the parameter being measured is a k-dimensional vector, then $\mathcal{P}$ is the Euclidean Space of dimension k.

iii) $\left\{\mathcal{E}_i\right\}_{i=1}^m$ is a collection of subsets of the environment space $\mathcal{X}$ such that $\bigcup_{i=1}^m \mathcal{E}_i = \mathcal{X}$. The collection $\left\{\mathcal{E}_i\right\}_{i=1}^m$ is called a *chart* on $\mathcal{X}$.(Fig. 2)

iv) $\{\mathcal{S}_i\}_{i=1}^{m}$ is a collection of m sensor-clusters, each $\mathcal{S}_i$ being assigned to an $\mathcal{E}_i$. Each $\mathcal{S}_i$ is a cluster of n_i abstract sensors $\mathcal{S}_i = \{\sigma_{i,j}^t\}_{j=1}^{n_i}$, where each abstract sensor $\sigma_{i,j}^t$ is a time-varying measurable function mapping the set $\mathcal{E}_i$ onto a subset $p_{i,j}^t$ of $\mathcal{X} \times \mathcal{P}$. Thus at any instant of time t, we have the collection of subsets $\{p_{i,j}^t\}_{j=1}^{n_i}$ of $\mathcal{X} \times \mathcal{P}$ as the abstract sensor estimates of the sensors $\{\sigma_{i,j}^t\}_{j=1}^{n_i}$ of the parameter value observed in $\mathcal{E}_i$ at time t.

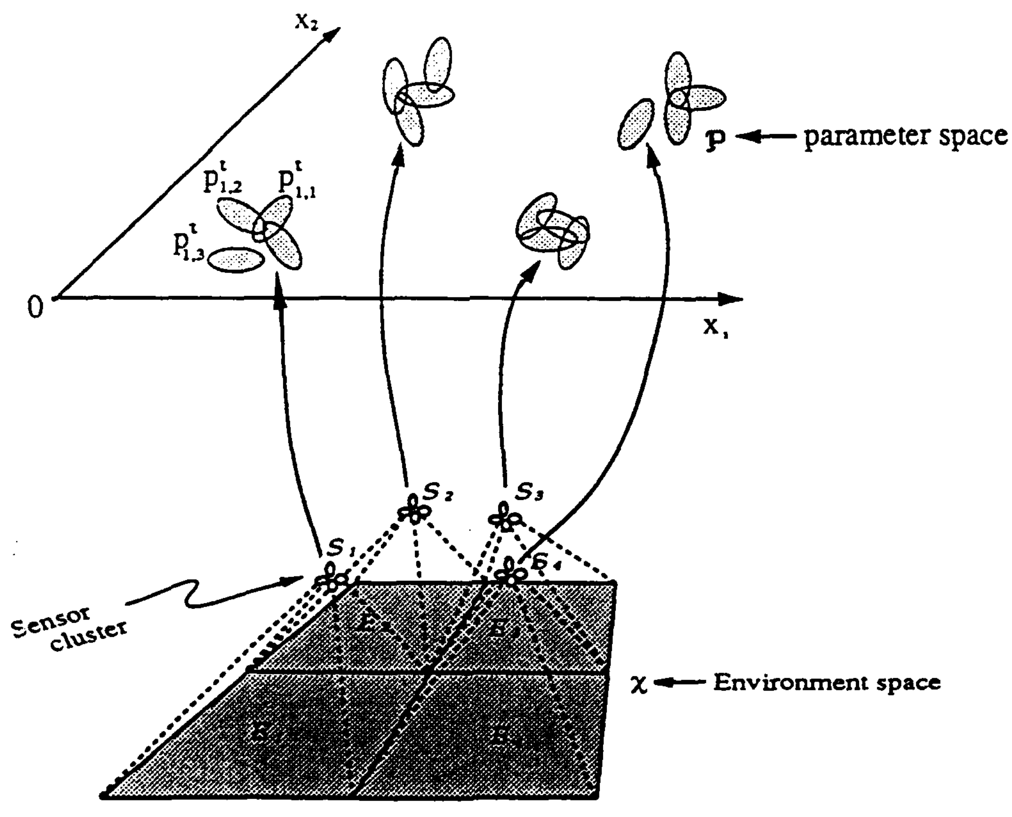

Fig 2. Schematic Representation of an abstract DSN

3.2 Illustrations and Remarks:

For example:

i) $\mathcal{E}_i$ may be regions in space and $\sigma^t_{i,j}$ may be sensors recording the position or velocities of moving objects in the region $\mathcal{E}_i$.

ii) The $\mathcal{E}_i$ may be regions in a terrain being mapped and $\sigma^t_{i,j}$ may be sensors recording the altitudes of various points of $\mathcal{E}_i$.

iii) The $\mathcal{E}_i$ may be regions in the interior of a boiler or a nuclear reactor and $\sigma^t_{i,j}$ may be sensors recording the temperature at various points of $\mathcal{E}_i$.

It is to be noted that each of the sensors $\sigma^t_{i,j}$ in the cluster $\mathcal{S}_i$ collects data from the same region in the environment space $\mathcal{X}$, namely $\mathcal{E}_i$. This redundancy is required to incorporate fault-tolerance in the distributed sensor network. Under ideal fault-free conditions, all the $\sigma^t_{i,j}$ in $\mathcal{S}_i$ have identical outputs. These sensors are abstract models of physical sensors, and since physical sensors can have uncertainties or be faulty, we need to incorporate these features into abstract sensors for realistic modelling. This is achieved by letting each sensor take a set of values in $\mathcal{P}$ at each point in the region $\mathcal{E}_i$ instead of a single value in $\mathcal{P}$. Thus if we denote by $p_{i,j}(x,t)$ the set of values corresponding to the sensor $\sigma^t_{i,j}$'s reading at the point $x \in \mathcal{E}_i$, then $p_{i,j}(x,t)$ is a subset of $\{x\} \times \mathcal{P}$, where $\{x\}$ is the singleton subset of $\mathcal{X}$ containing the point x. Of course, $p_{i,j}(x,t)$ is contained in $p^t_{i,j}$ (see Fig. 3). Thus the outputs of the sensors of a cluster $\mathcal{S}_i$ representing the value of the parameter being measured at the point $x \in \mathcal{E}_i$ at time t are subsets of $\{x\} \times \mathcal{P}$.

These are to be integrated to obtain a single subset which is a fault-tolerant estimate of the parameter value measured at the point x at time t. The definition of a correct sensor crucial for this is given in the next section.:

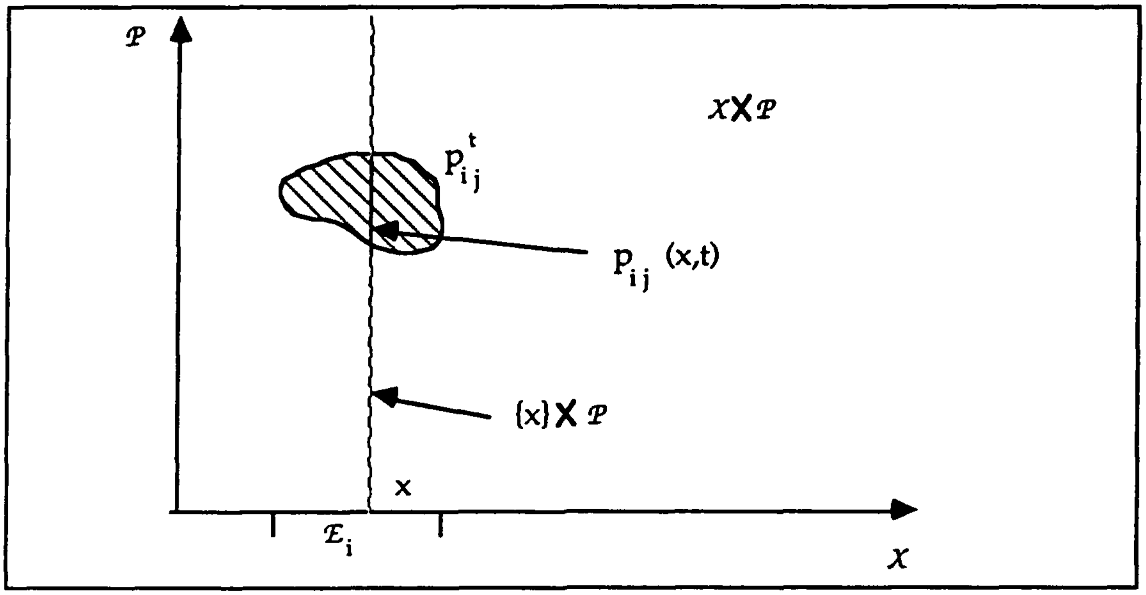

Fig. 3: The subsets $p^t_{i,j}$ and $p_{i,j}(x,t)$ corresponding to the sensor $\sigma^t_{i,j}$

The subsets $p^t_{i,j}$ and $p_{i,j}(x,t)$ corresponding to the sensor $\sigma^t_{i,j}$ are shown above in the case when both the spaces $\mathcal{X}$ and $\mathcal{P}$ are 1-dimensional.

The subset $p_{i,j}(x,t)$ is a line segment. If $S^t_{i,j}$ is correct at x at time t, then the point (x, a) lies on the line segment $p_{i,j}(x,t)$ where a is the correct value of the parameter at x and at time t.

In general, if $\mathcal{P}$ is k-dimensional, then $p_{i,j}(x,t)$ is generally a k-dimensional hypercuboid where the length of each side is a measure of the uncertainty in the corresponding component of the k-dimensional parameter being measured.

3.3 Fault -tolerant integration of abstract sensor outputs

We now describe a computational characterization of sensor integration for general distributed sensor networks. We define correctness and faultiness in abstract sensors as follows:

<u>Definition 2:</u> An abstract sensor $S^{t}_{i,j} \in \mathcal{S}_i$ is *correct* at $x \in \mathcal{E}_i$ at time t if $(x, a) \in p_{i,j}(x,t)$ where a is the actual value of the parameter being measured at x at time t ($a \in \mathcal{P}$), else it is faulty..

The method of integration of our ealier paper[LIKM 91] can be used here to obtain a fault-tolerant estimate of the actual physical value being measured in this general case also as shown blow:

Consider the characteristic function of the set $p_{i,j}(x,t)$:

$$\chi^{i,j}_{x,t}(y) = \begin{cases} 1 & \text{if } y \in p^{t}_{i,j} \\ 0 & \text{otherwise} \end{cases}$$

If atmost f_i of the n_i sensors in the cluster $\mathcal{S}_i$ are known to be faulty, then following the method of integration employed in our earlier paper[LIKM 91], the correct value of the parameter being measured by the sensor cluster s_i at $x \in \mathcal{E}_i$ and at time t is contained in the subset corresponding to the characteristic function $\chi^{i}_{x,t}(y)$ given by

$$\chi^{i}_{x,t}(y) = \chi_{[n_i - f_i, \infty)}(O^{i}_{x,t}(y))$$

where $$\chi_{[a,\infty)}(x) = \begin{cases} 1 & \text{if } x \in [a, \infty) \\ 0 & \text{otherwise} \end{cases}$$

and $O^i_{x,t}(y) = \sum_{j=1}^{n_i} \mathcal{X}^{i,j}_{x,t}(y)$ is the overlap function for the cluster $\mathcal{S}_i$ at the point $x \in \mathcal{E}_i$ and at time t. i.e. $O^i_{x,t}(y)$ gives the number of sensors $\sigma^t_{i,j}$ whose estimates $p_{i,j}(x,t)$ at the point $x \in \mathcal{E}_i$ and at the time t contain the point (x, y) $\in \{x\} \times \mathcal{P}$, $y \in \mathcal{P}$.

Consequently, $\mathcal{X}^i_{x,t}(y)$ is the characteristic function of the set of all those points of $\{x\} \times \mathcal{P}$ which lie in the intersection of at least n_i-f_i intersections of the sets $p_{i,j}(x,t)$.

The correct value of the parameter being measured at $x \in \mathcal{E}_i$ at time t must lie in the subset $\Sigma^i(x, t) = \{ y \in \mathcal{P} \mid \mathcal{X}^i_{x,t}(y) = 1\}$, since the assumption that atmost f_i sensors are faulty implies at least n_i-f_i sensors are correct and hence must overlap since their estimates $p_{i,j}(x,t)$ must all contain the point (x, a) where a is the actual value of the parameter being measured.

Faulty sensors could have random or "wild" faults, in which event there is little correlation between the sensor's estimate and the correct physical value of the parameter being measured. On the other hand, the fault may be "tame", in which case the faulty sensor's estimate although does not contain the actual physical value of the parameter being measured, lies sufficiently close to it. It is reasonable to assume that most sensors fail tamely due to perturbations encountered in the physical world, in which case faulty sensors tend to cluster in the neighbourhood of the correct value of the parameter. We may make use of this clustering to predict with reliability the subset with highest chance of containing the actual value of the parameter among those subsets which belong to $\sum^i(x, t)$. In order to do this, we need to make the notion of a tame fault more rigourous:

Definition3: A faulty sensor is *tamely* faulty at x at time t if it intersects with a sensor that is correct at x at time t.

3.4 Reduction of the Output measure when most faults are tame

It is unlikely that sensors with wild or random faults cluster since by their very nature, they are uncorrelated and hence distributed more or less evenly. In the case when most sensor faults are tame, we may resort to the polling technique introduced in an earlier paper[LIKM 91] to reduce the measure of the subset containing the correct physical value with high reliability:

Let $L^i_1(x, t), \ldots, L^i_{k_i}(x, t)$ be the disjoint maximal connected subsets of $\mathcal{P}$ whose union is the subset $\sum^i(x, t)$ containing the correct physical value of the parameter measured at x at time t. ($x \in \mathcal{E}_i$).

Define the *popularity* of the output of the k^{th} sensor($1 \leq k \leq n_i$) in the cluster $\mathcal{S}_i$ to be the nonnegative integer $\pi^{ik}(x, t)$ given by:

$$\pi^{ik}(x, t) = (\sum_{j=1}^{n_i} || \mathcal{X}^{ij}_{x,t} \mathcal{X}^{ik}_{x,t} ||) - 1$$

where $||f||$ is the maximum value of the function f. $\pi^{ik}(x, t)$ gives the number of sensor outputs having nonemply intersection with the output of the sensor $\sigma^{t}_{i,k}$ at x and time t.

If $\Lambda^{ij}_{x,t}$ is the characteristic function of the set $L^{i}_{j}(x, t)$, $(1 \leq j \leq k_i)$ then define the reliability $R^{ik}(x, t)$ of the set $L^{i}_{j}(x, t)$ to be the nonnegative integer given by:

$$R^{ik}(x, t) = \sum_{k=1}^{n_i} ||\Lambda^{ij}_{x,t} \chi^{ik}_{x,t}||\pi^{ik}(x, t)$$

$R^{ik}(x, t)$ is the sum of the popularities of all those sensors' outputs which contain the set $L^{i}_{j}(x, t)$,.

It is clear that the larger the reliability of the connected set $L^{i}_{j}(x, t)$,, the greater the clustering of sensors about $L^{i}_{j}(x, t)$, and hence the greater the likelihood of $L^{i}_{j}(x, t)$, containing the correct value of the parameter. Thus $R^{ij}(x, t)$ is a good measure of the connected subsets $L^{i}_{j}(x, t)$ of $\sum^{i}(x, t)$ containing the correct value of the parameter. (It is obvious that one and only one of these maximal connected subsets of $\sum^{i}(x, t)$ contains the correct parameter value). The maximal connected subsets with the highest reliability may be chosen to represent the estimate of the sensor cluster $\mathcal{S}_i$ This analysis greatly reduces the measure of the output subset of $\mathcal{S}_i$ and is an efficient fault-tolerant integration technique for the cas' when most sensor faults are tame.Thus, it is seen that the techniques employed by us in the paper[LIKM 91] though used on 1-dimensional interval estimates, hold for a much more general class of sensor outputs.

4.0 Additional Remarks

If the sensors provide additional information on their outputs (e.g. a probability distribution or a weight function on their output values), then we may replace the characteristic function of the output sets by these function and look for an appropriate integration method which combines these functions (in such a way that the "correctness" of the output is not sensitive to a small number of faults in the sensor cluster) to give a function whose domain contains the correct value of the parameter measured by the cluster, and the correct value lies in a high weighted or high probability region of this new function/distribution. Depending on the kind of extra information provided by the sensors and the functions involved, the method of integration of these functions varies, and it is our future goal to make an in-depth study of certain useful integration techniques for various kinds of sensor outputs with probability or weight-functions attached.

All of the above analysis is local to each subset $\mathcal{E}_i$ of the chart on the environment space X.

The integration of the sensor outputs in each cluster is done by the processor allocated to that cluster. Thus the parameter measured in each subset of the chart on X is measured and estimated by the sensor cluster-processor unit of that subset. The subsets of the chart on X form a tiling of the space X. If the sensors are probes capable of measuring a parameter's value only at a given point then X will be the collection of all these points. If the sensors are capable of sampling a region, then the space X is the union of all these regions. The sampling of the sensor outputs are done either periodically in time or randomly. But they are all sampled at the same instant of time, for otherwise integration of the sensor values would not make sense. We may then proceed to get an over-all picture of the behaviour of the parameter over X by studying the outputs of all the sensor clusters. This is done by appropriately juxtaposing the sensor values according to the actual geometric layout of the chart and obtaining the profile of the parameter over the space over which the sensor clusters are distributed.

It is also possible that the processors allocated to the sensor clusters may have to communicate among themselves for efficient over-all performance. For example, in the case when the sensors are tracking a moving object, the object moves from one set in the chart to another, the sensor cluster currently observing the object activates the neighbouring sensor cluster which prepares to track the object from an indicated position on the boundary of the chart set it is allocated to. Here, one may take advantage of this kind of inter-processor communication to check for faults by comparing common values. It is therefore often fruitful to over lap the sets in the chart to obtain extra redundancy and greater fault-tolerance.

5.0 A SCHEME FOR PATCHING TOGETHER LOCAL PROCESSOR INFORMATION TO OBTAIN A GLOBAL PICTURE OF THE PARAMETER MEASURED.

We now look at the problem of addressing the global behaviour of a parameter over the space $\mathcal{X}$. This involves the interaction of the processors allocated to the subsets of the chart and comparison of common information for smooth patching of local information to obtain a global picture of the phenomenon being observed over $\mathcal{X}$.

Each of the subsets of the subsets $\mathcal{E}_i$ $(1 \leq i \leq m,\ \bigcup_{i=1}^{m} \mathcal{E}_i = \mathcal{X})$ covers a region of the environment space $\mathcal{X}$, and is equipped with a sensor-cluster $\mathcal{S}_i = \left\{\sigma_{i,j}^{t}\right\}_{j=1}^{n_i}$, where each sensor $\sigma_{i,j}^{t}$ monitors all of $\mathcal{E}_i$ and sends data to a common processor allocated to $\mathcal{E}_i$. Each $\mathcal{E}_i$ is equipped with a coordinate system of its own, and all the sensors in the cluster $\mathcal{S}_i$ measure with reference to this coordinate system. Thus all points x in $\mathcal{E}_i$ have local coordinates and if two of the region $\mathcal{E}_i$ overlap then the coordinates of a point in the intersection will be different in the different regions containing it. The relative arrangement of the subsets $\mathcal{E}_i$ of the chart on $\mathcal{X}$ depends upon the specific needs of a distributed sensor network. However, it is desirable to have them overlapping since this helps in patching up the local scenes to form a global picture as well as in increasing fault-tolerant and aiding fault-detection in sensors by comparison of data at common points.

If $\mathcal{E}_i$ and $\mathcal{E}_j$ are two sets in the chart on $\mathcal{X}$ that overlap and a point x in the intersection has local coordinates $(x_1^i, .., x_k^i)$ and $(x_1^j, .., x_k^j)$ in the sets $\mathcal{E}_i$ and $\mathcal{E}_j$ respectively then we can obtain one set of local coordinates from the other by coordinate transformation.

That is, if $x \in \mathcal{E}_i \cap \mathcal{E}_j$ with local coordinates in $\mathcal{E}_i$ and $\mathcal{E}_j$ and as above, then we have a transformation T_{ij} defined on $\mathcal{E}_i \cap \mathcal{E}_j$ which transforms the local coordinates of x w.r.t $\mathcal{E}_i$ to the local coordinates of x w.r.t $\mathcal{E}_j$:

$$T_{ij}(x_1^i, .., x_k^i) = (x_1^j, .., x_k^j)$$

It is clear that the transformation T_{ij} and T_{ji} are inverses of each other on $\mathcal{E}_i \cap \mathcal{E}_j$.

For instance, the $\mathcal{E}_i$ may be k-dimensional hypercuboids of identical dimensions overlapping in some manner, and with cartesian local coordinates. Then the T_{ij} are affine linear transformation involving a rotation and a translation.(see Fig. 4).

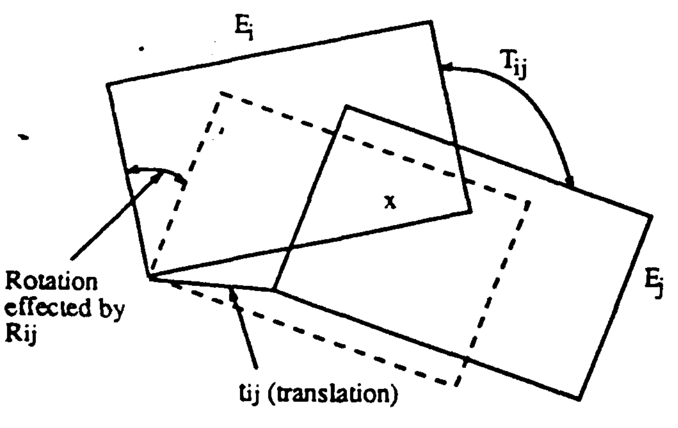

Fig 4. Linear transformation involving a rotation and a translation for T_{ij}

i.e, there exists a $k \times k$ matrix R_{ij} and a k-dimensional column vector t_{ij}, such that

$$(x)_j = R_{ij} \bullet (x)_i + t_{ij}$$

where $(x)_j$ and $(x)_i$ are column vector representations of the local coordinates of x in $\mathcal{E}_j$ and $\mathcal{E}_i$ respectively.

i.e. $T_{ij}(x)_i = R_{ij} \bullet (x)_i + t_{ij}$

It is also possible that the $\mathcal{E}_i$ are not rigidly fixed but move in time and space as is the case with tracking sensor systems. In this case, the transformations T_{ij} are time-varying. In the case when there is only one chart set and it is in motion, then the time-varying transformation which allows for the movement of the sensor system corresponds to an inertial navigation system. A chart equipped with local coordinates is called a coordinate chart on $\mathcal{X}$, and the transformation that effect local coordinate changes are called transition functions.

The transition functions are known beforehand if the chart sets are fixed. If the chart sets move in a predictable or prescribed way then again the time-varying transition function can be computed.

Examples:

The following two figures are simple examples of overlapping chartsets of 2-dimensional environment spaces and the transition functions are simple translations or rotations:

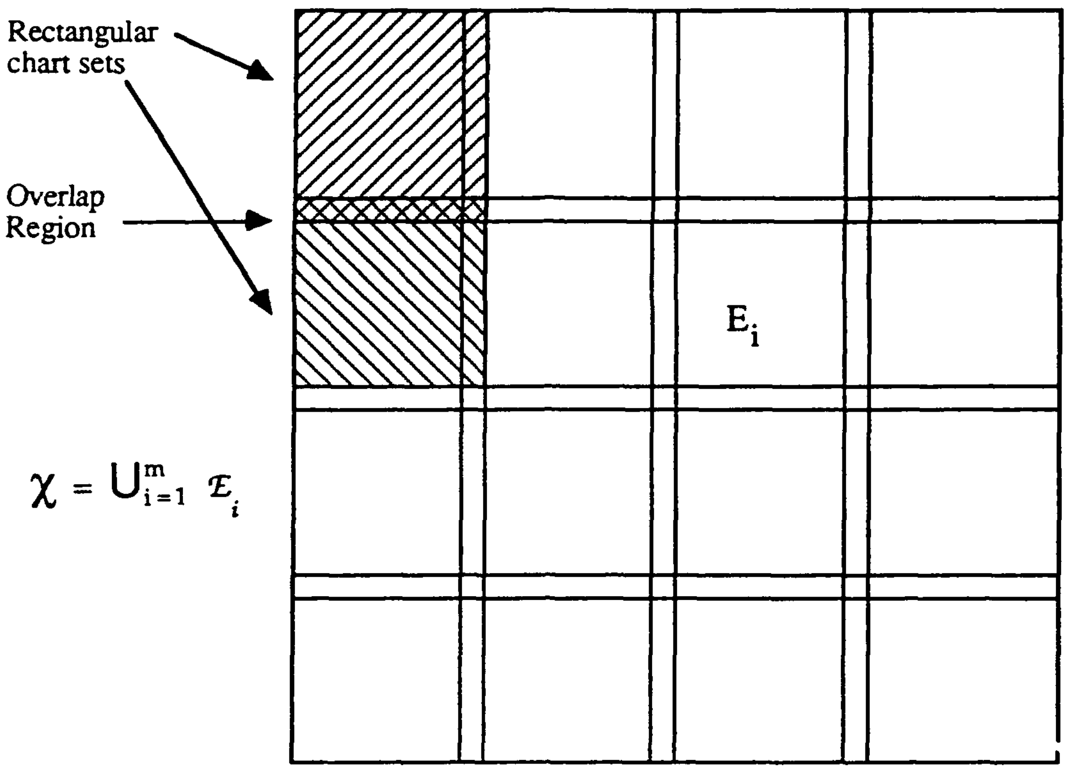

Fig. 5. A 2-Dimensional rectangular region X tiled by partially overlapping rectangular chart sets. The transition functions here are just horizontal or vertical translations.

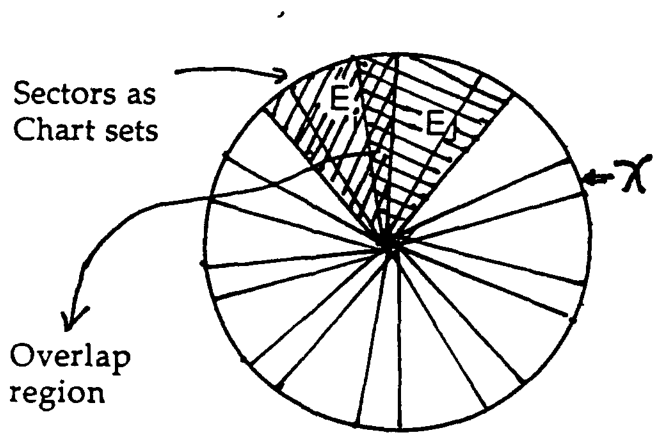

Fig 6. A 2-dimensional circular region X tiled by partially overlapping sectors. The transition functions here are clockwise and counter-clockwise rotations.

Further analysis in this direction can be done in specific distributed sensor network situation, but the general interprocessor interaction and communication must involve the use of transition function for meaningful comparison of data. The coordinated changes are effected by local processors and only those processors that are connected have transition functions defined. In the case of moving chart sets it is possible that two sets whose processors are not connected directly may overlap for some time. It is possible to perform local coordinate changes by routing data through other intermediate processors since the transition functions are transitive and hence may be composed to obtain transformations between unconnected processors.

6.0 CONCLUDING REMARKS

The paradigm discussed above is very general, and subsumes a wide range of distributed sensor networks. Study of fluid flow and fluid thermodynamics in conduits, temperature monitoring in nuclear reactors, detecting and tracking targets in air, sonar tracking of submarines, sensing and detecting in meteorology, atmospheric studies and Oceanography are a few of the many applications which would find the above paradigm not only useful but also natural for formulating and designing a distributed sensor network. We intend to explore further the application and implementation of the paradigm with emphasis on the algorithmic aspects of sensor integration.

7.0 REFERENCES

[BlBr 90] Blackman, S.S., and Broida, T.J., "Multiple Sensor Data Association and Fusion in Aerospace Applications", J. of Robotic systems, 7(3), 445-485 (1990).

[Duwh 88] Durrant-Whyte, H.F., "Sensor Models and Multisensor Integration," Int. J. Robot. Res., V. 7, No. 6, 1988, pp. 97-113.

[GeCh 90] Geraniotis E. and Chau Y.A., "Robust Data Fusion for Multisensor Detection Systems," IEEE Transactions on Information Theory, Vol. 36., No. 6, Nov. 1990, pp1265-1279.

[HAMD 87] Henderson, T.C., Allen, P.K., Mitchie, A., Durrant-Whyte, H., Snyds, W., Eds., "Workshop on Multisensor Integration in Manufacturing Automation," Dept. of Comp. Scie., Univ. of Utah, Snowbird, Tech. Rpt. UUCS-87-006, Feb. 1987.

[Hube 81] "Robust Statistics," Wiley, New York, 1981.

[KaOM 90] R.L.Kashyap, S.G. Oh and R.N.Madan, "Robust Decentralized Direction of Arrival Estimation in Contaminated Noise," IEEE Trans. ASSP, Vol. 38, No. 3, 1990, pp.496-505.

[LIKM 91] Lakshman Prasad and S.S.Iyengar, R.L.Kashyap and R.N. Madan, "Functional Characterization of sensor integration in distributed sensor networks." To appear in IEEE-SMC.

[LuKa 89] Luo, R.C. and Kay, M.G., "Multisensor Integration and Fusion in Intelligent Systems," IEEE Trans. on SMC, V. 19, No. 5, 1989, pp. 901-931.

[LuLS 88] Luo, R.C., Lin, M. and Scherp, R.S., "Dynamic Multi-sensor Data Fusion System for Intelligent Robots," IEEE J. Robot. Automat., V. RA-4, No. 4,1988, pp. 386-396.

[Marz 89] Marzullo, K, "Implementing Fault-Tolerant Sensors", TR89-997, Department of Computer Science, Cornell University, May 1989.

[Zhen 89] Zheng, Y.F., "Integration of Multiple Sensors into a Robotic System and its Performance Evaluation," IEEE Trans. on Robot. Automat., V. 5, No. 5, Oct. 1989, pp. 658-669.