Sensor Fusion

Thomas C. Henderson[1], Mohamed Dekhil[1], Robert R. Kessler[1], and Martin L. Griss[2]

[1] Department of Computer Science, University of Utah, USA
[2] Hewlett Packard Labs, USA

Sensor fusion involves a wide spectrum of areas, ranging from hardware for sensors and data acquisition, through analog and digital processing of the data, up to symbolic analysis all within a theoretical framework that solves some class of problem. We review recent work on major problems in sensor fusion in the areas of theory, architecture, agents, robotics, and navigation. Finally, we describe our work on major architectural techniques for designing and developing wide area sensor network systems and for achieving robustness in multisensor systems.

1. Introduction

Multiple sensors in a control system can be used to provide:

- *more information*,
- *robustness*, and
- *complementary information*.

In this chapter, we emphasize the first two of these. In particular, some recent work on wide area sensor systems is described, as well as tools which permit empirical performance analysis of sensor systems.

By *more information* we mean that the sensors are used to monitor wider aspects of a system; this may mean over a wider geographical area (e.g. a power grid, telephone system, etc.) or diverse aspects of the system (e.g. air speed, attitude, acceleration of a plane). Quite extensive systems can be monitored, and thus, more informed control options made available. This is achieved through a higher level view of the interpretation of the sensor readings in the context of the entire set.

Robustness has several dimensions to it. First, statistical techniques can be applied to obtain better estimates from multiple instances of the same type sensor, or multiple readings from a single sensor [15]. Fault tolerance is another aspect of robustness which becomes possible when replacement sensors exist. This brings up another issue which is the need to monitor sensor activity and the ability to make tests to determine the state of the system (e.g. camera failed) and strategies to switch to alternative methods if a sensor is compromised.

As a simple example of a sensor system which demonstrates all these issues, consider a fire alarm system for a large warehouse. The sensors are widely dispersed, and, as a set, yield information not only about the existence of a fire, but also about its origin, intensity, and direction of spread. Clearly, there is a need to signal an alarm for any fire, but a high expense is incurred for false alarms. Note that complementary information may lead to more robust systems; if there are two sensor types in every detector such that one is sensitive to particles in the air and the other is sensitive to heat, then potential non-fire phenomena, like water vapor or a hot day, are less likely to be misclassified.

There are now available many source materials on multisensor fusion and integration; for example, see [1, 3, 14, 17, 24, 28, 29, 32, 33], as well as the bi-annual IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems.

The basic problem studied by the discipline is to satisfactorily exploit multiple sensors to achieve the required system goals. This is a vast problem domain, and techniques are contingent on the sensors, processing, task constraints, etc. Since any review is by nature quite broad in scope, we will let the reader peruse the above mentioned sources for a general overview and introduction to multisensor fusion.

Another key issue is the role of control in multisensor fusion systems. Generally, control in this context is understood to mean control of the multiple sensors and the fusion processes (also called the multisensor fusion architecture). However, from a control theory point of view, it is desirable to understand how the sensors and associated processes impact the control law or system behavior. In our discussion on robustness, we will return to this issue and elaborate our approach. We believe that robustness at the highest level of a multisensor fusion system requires adaptive control.

In the next few sections, we will first give a review of the state of the art issues in multisensor fusion, and then focus on some directions in multisensor fusion architectures that are of great interest to us. The first of these is the revolutionary impact of networks on multisensor systems (e.g. see [45]), and Sect. 3. describes a framework that has been developed in conjunction with Hewlett Packard Corporation to enable enterprise wide measurement and control of power usage. The second major direction of interest is robustness in multisensor fusion systems and we present some novel approaches for dealing with this in Sect. 4. As this diverse set of topics demonstrates, multisensor fusion is getting more broadly based than ever!

2. State of the Art Issues in Sensor Fusion

In order to organize the disparate areas of multisensor fusion, we propose five major areas of work: theory, architectures, agents, robotics, and navigation. These cover most of the aspects that arise in systems of interest, although there is some overlap among them. In the following subsections, we highlight topics of interest and list representative work.

2.1 Theory

The theoretical basis of multisensor fusion is to be found in operations research, probability and statistics, and estimation theory. Recent results include methods that produce a minimax risk fixed-size confidence set estimate [25], select minimal complexity models based on Kolmogorov complexity [23], use linguistic approaches [26], tolerance analysis [22, 21], define information invariance [11], and exploit genetic algorithms, simulated annealing and tabu search [6]. Of course, geometric approaches have been popular for some time [12, 18]. Finally, the Dempster-Shafer method is used for combining evidence at higher levels [30].

2.2 Architecture

Architecture proposals abound because anybody who builds a system must prescribe how it is organized. Some papers are aimed at improving the computing architecture (e.g. by pipe-lining [42]), others aim at modularity and scalability (e.g. [37]). Another major development is the advent of large-scale networking of sensors and requisite software frameworks to design, implement and monitor control architectures [9, 10, 34]. Finally, there have been attempts at specifying architectures for complex systems which subsume multisensor systems (e.g. [31]). A fundamental issue for both theory and architecture is the conversion from signals to symbols in multisensor systems, and no panacea has been found.

2.3 Agents

A more recent approach in multisensor fusion systems is to delegate responsibility to more autonomous subsystems and have their combined activity result in the required processing. Representative of this is the work of [5, 13, 40].

2.4 Robotics

Many areas of multisensor fusion in robotics are currently being explored, but the most crucial areas are force/torque [44], grasping [2], vision [39], and haptic recognition [41].

2.5 Navigation

Navigation has long been a subject dealing with multisensor integration. Recent topics include decision theoretic approaches [27], emergent behaviors [38], adaptive techniques [4, 35], frequency response methods [8]. Although the majority of techniques described in this community deal with mobile robots, there is great interest in applying these approaches to riverboats, cars, and other modes of transportation.

We will now present some very specific work relating to wide area sensor networks and robustness of multisensor systems.

3. Wide Area Sensor Networks

In collaboration with Hewlett-Packard Laboratories (HPL), we have been developing an experimental distributed measurement software framework that explores a variety of different ways to make the development of Distributed Measurement and Control (DMC) systems easier. Our technology offers the ability to rapidly develop, deploy, tune, and evolve complete distributed measurement applications. Our solution makes use of transducers attached to the HP Vantera Measurement and Control Nodes for DMC [7]. The software technology that we have developed and integrated into our testbed includes distributed middleware and services, visual tools, and solution frameworks and components. The problem faced here is that building robust, distributed, enterprise-scale measurement applications using wide area sensor networks has high value, but is intrinsically difficult. Developers want enterprise-scale measurement applications to gain more accurate control of processes and physical events that impact their applications.

A typical domain for wide area sensor networks is energy management. When utilities are deregulated, more precise management of energy usage across the enterprise is critical. Utilities will change utility rates in real-time, and issue alerts when impending load becomes critical. Companies can negotiate contracts for different levels of guaranteed or optional service, permitting the utility to request equipment shut off for lower tiers of service. Many Fortune 500 companies spend tens of millions of dollars each year on power, which could change wildly as daily/hourly rates start to vary dynamically across the corporation. Energy costs will go up by a factor of 3 to 5 in peak load periods. Measurement nodes, transducers and controllers distributed across sites and buildings, will be attached to power panels and which enable energy users to monitor and control usage. Energy managers at multiple corporate sites must manage energy use and adjust to and balance cost, benefit and business situation. Site managers, enterprise workflow and measurement agents monitor usage, watch for alerts and initiate load shifting and shedding (see Fig. 3.1).

The complete solution requires many layers. Data gathered from the physical processes is passed through various information abstraction layers to provide strategic insight into the enterprise. Likewise, business level decisions are passed down through layers of interpretation to provide control of the processes. Measurement systems control and access transducers (sensors and actuators) via the HP Vantera using the HP Vantera Information Backplane publish/subscribe information bus. Some transducers are self-identifying and provide measurement units and precision via Transducer Electronic Data Sheets (TEDS - IEEE 1451.2).

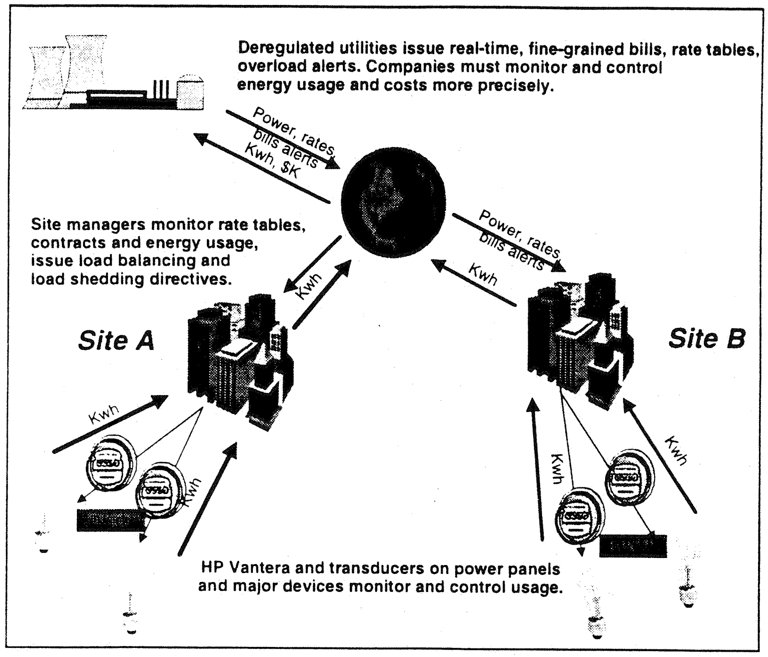

Fig. 3.1. Energy management

3.1 Component Frameworks

How are applications like this built? Enterprise and measurement applications are built from a set of components, collectively called frameworks. Each framework defines the kinds of components, their interfaces, their services, and how these components can be interconnected. Components can be constructed independently, but designed for reuse. In a distributed environment, each component may be able to run on a different host, and communicate with others [16].

For this testbed, we have developed the (scriptable) Active Node measurement oriented framework, using the Utah/HPL CWave visual programming framework as a base [34]. This framework defines three main kinds of measurement components: (1) Measurement Interface Nodes that provide gateways between the enterprise and the measurement systems, (2) Active Nodes that provide an agency for measurement abstraction, and (3) Active Leaf Nodes that act as proxies to monitor and control transducers. Each of these nodes communicates with the other types and the measurement devices.

The key component for the CWave measurement agency component model is the Active Node. Active nodes allow a measurement engineer to write scripts which communicate with other nodes via the HP Vantera publish/subscribe information bus and thus control component interactions. The scripts run via the Microsoft ActiveX Scripting engine.

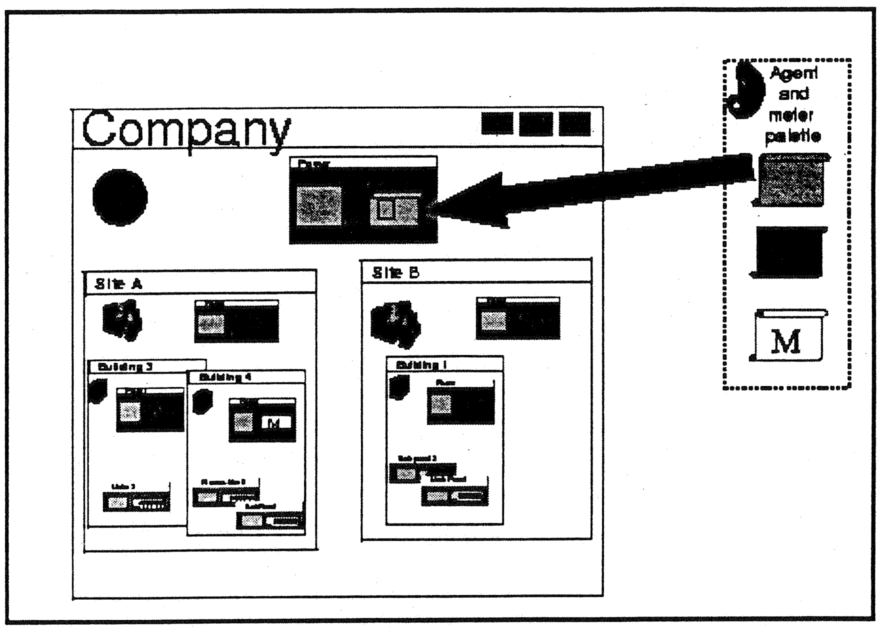

Fig. 3.2. Basic structure of agent and analysis components

One or more agent scripts, written in VBScript, JavaScript, or Perl can be downloaded into the component (by drag and drop). Once downloaded, each script runs its own execution thread and can access the full ActiveX scripting engine. These scripts can publish new topics, subscribe to subjects, and interact with the CWave environment. The Measurement Interface Nodes (MIN) act as gateways between the HP Vantera Information Bus and the enterprise world.

CWave is a uniquely *visual* framework, consisting of components and tools that are targeted for quickly building, evolving, testing and deploying distributed measurement systems. CWave features include extensibility, visual development of scripts, downloaded programs, multi-machine visualization and control, drag-and-drop, and wiring. Active Nodes are instances of a CWave component that has been specially interfaced to the HP Vantera environment, and make heavy use of the publish/subscribe capabilities.

In Fig. 3.2 CWave components are shown which represent buildings and sites. This was built by dragging components from the palette, customizing the names, and setting properties from the property sets. Detail can be suppressed or revealed via zooming and selective viewing. Nesting of components provides a natural namespace for components that is important in controlling the scope of publish/subscribe. We also show a palette of *agent scripts* that can be dragged and dropped onto an Active Node or Active Leaf Node. The scripts can then be propagated to children. The CWave environment can be used by engineers to develop components and skeletal applications, by installers to configure and customize a system, or by the end user engineers to adjust measurements, to monitor processing, etc.

The deployed set of agent scripts work together to collect the power data from several sensors, and combine these into an average energy measurement over the interval. These measurements are propagated up, with averaging, and tests for missing measurements to ensure robustness of the result. Other agent scripts can be deployed to monitor or log data at various Active Nodes, or to test for conditions and issue alerts when certain (multi-sensor) data conditions arise.

CWave provides visual programming, threads and distributed management. The addition of the Active Node framework provides a flexible and customizable model for combining and integrating the inputs and control of a multitude of sensors.

4. Robustness

Multisensor fusion techniques have been applied to a wide range of problem domains, including: mobile robots, autonomous systems, object recognition, navigation, target tracking, etc. In most of these applications, it is necessary that the system perform even under poor operating conditions or when subcomponents fail. We have developed an approach to permit the developer to obtain performance information about various parts of the system, from theory, simulation or actual execution of the system. We have developed a semantic framework which allows issues to be identified at a more abstract level and then monitored at other levels of realization of the system.

Our overall goal is shown in Fig. 4.1. A system is comprised of software, hardware, sensors, user requirements, and environmental conditions. A system model can be constructed in terms of models of these components. An analysis can be done at that level, but is usually quite abstract and of a worst-case kind of analysis. We want to use such models to gain insight into where in a simulation or actual system to put taps and monitors to obtain empirical performance data. These can be used for several purposes:

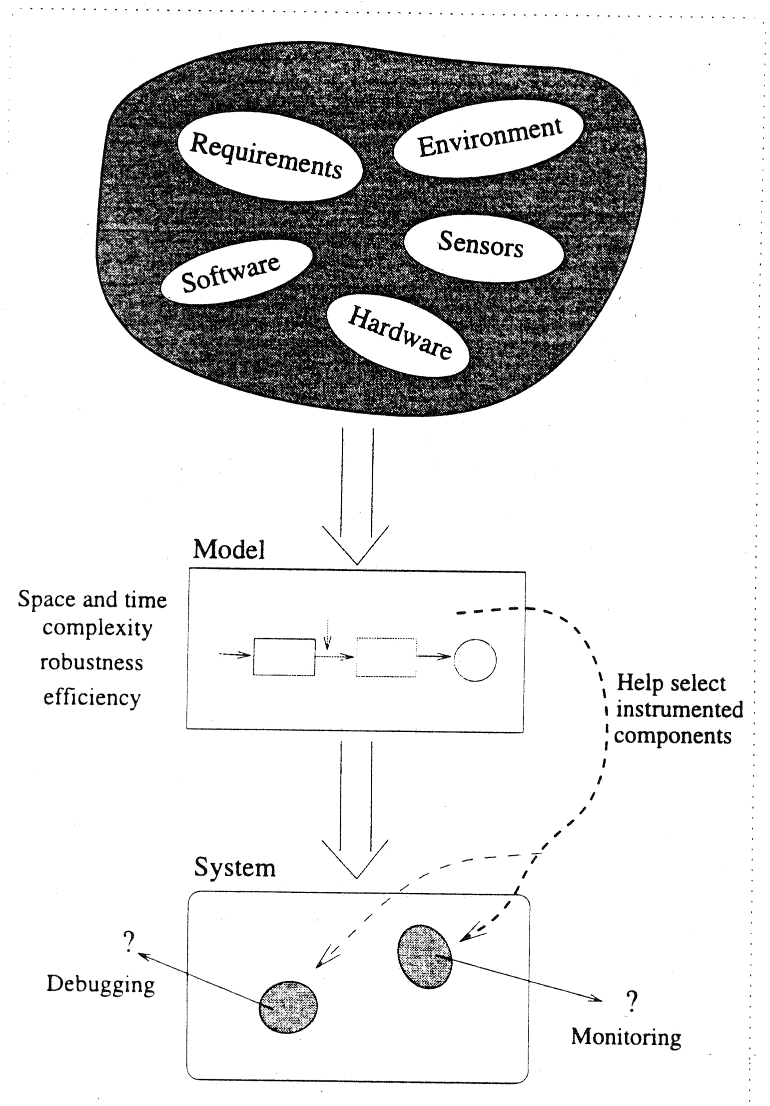

Fig. 4.1. The proposed modeling approach

- *Debugging*: the designer can interactively watch important aspects of the system as it runs.
- *Design Choices*: the designer may want to measure the performance of alternative solutions over various domains (time, space, error, etc.).
- *Adaptive Response*: the designer may want to put in place monitors which watch the performance of the system, or its context, and change techniques during execution.

This allows robustness to be built into a system, and the user can do so on well-founded information. For a more detailed discussion of this approach, see [9, 10] where we applied the framework to the performance comparison of a visual technique and a sonar technique for indoor wall pose estimation.

4.1 Instrumented Sensor Systems

Since we are putting in functions to monitor the data passing through a module and its actions, we call our approach *Instrumented Sensor Systems*. Figure 4.2 shows the components of our framework. As shown in the figure,

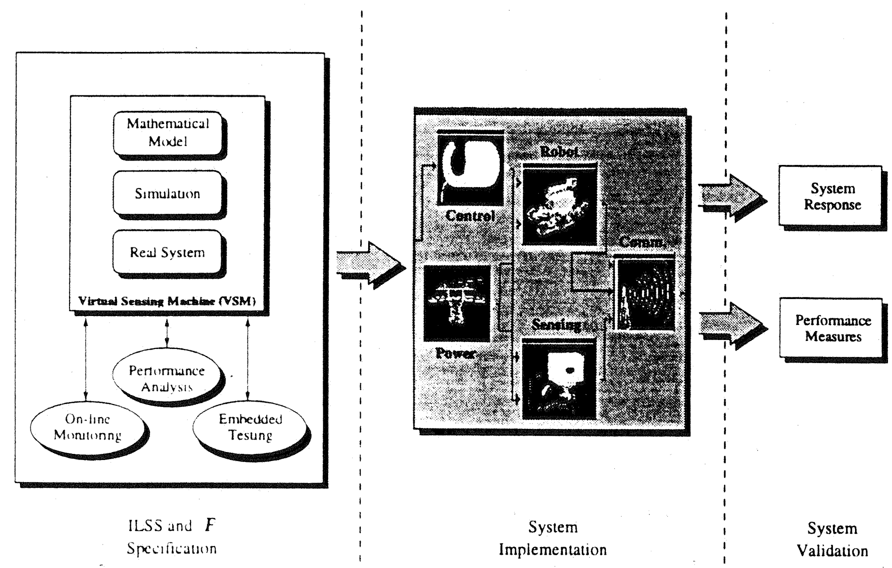

Fig. 4.2. The instrumented logical sensor system components

this approach allows for a specification, an implementation, and the validation of results. This is an extension of our previous work on *Logical Sensor Specifications* [19, 20]. A logical sensor is basically a system object with a name, characteristic (typed) output, and a select function which chooses between various alternate methods for producing the desired data. An *Instrumented Logical Sensor* has the following additional functions:

- *Taps*: Provide for a trace of the flow of data along a component connection path,
- *Tests*: Run the currently selected method on a known input/output pair and check for correctness.
- *Monitors*: Check for failure or adaptive mode conditions (monitors may run tests as part of their monitoring activity).

We have applied these ideas to the development of mobile robot systems. (Also, see [46] which influenced our work.)

4.2 Adaptive Control

Another area we are currently exploring is adaptive control. Figure 4.3 shows a basic feedback control loop. In a stable environment such an approach is

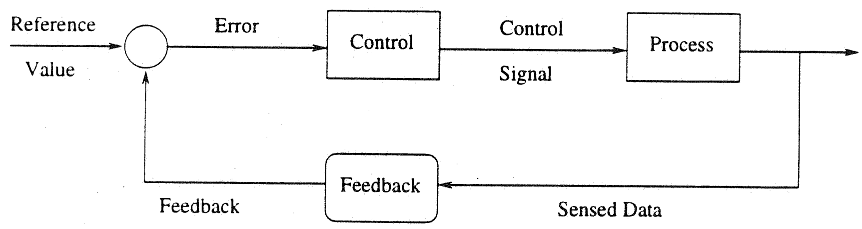

Fig. 4.3. Simple feedback control loop

feasible and many standard techniques can be applied (e.g. Kalman filtering is often used in navigation). Our concern is with a higher level of robustness — namely, when the context changes or assumptions fail, methods must be provided so the system can detect and handle the situation (see Fig. 4.4).

Our view is that sensor data can be used to alter the parameters of the control function. For example, consider a control function which solves for a variable (say, V which is a voltage to be applied to a motor), at each instant by determining where the following equation is 0:

$$\frac{dV}{dt} = a * (1 - \frac{V}{b}) - \frac{V}{(1 + V^2)} = f(V) \tag{4.1}$$

The parameters a and b represent features of the environment, and if sensors are used to determine them, then our function is really $f(V; a, b)$. In this case when the parameters change value, the solution space may change

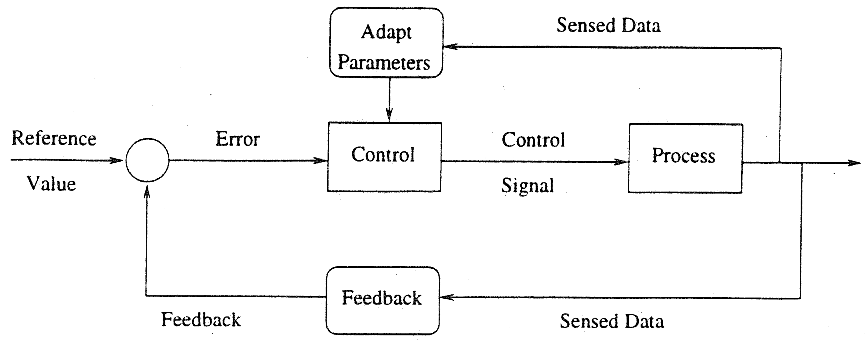

Fig. 4.4. Adaptive control

qualitatively; e.g. when $a = 0.2$ and $b = 10$, then there is a single solution (see Fig. 4.5). However, as the parameter a changes, there may be more than one solution (see Fig. 4.6). Finally, if systems are designed exploiting these features, then there may also be large jumps in solution values; Fig. 4.7 shows how with a slight increase in the value of a, the solution will jump from about 1.3 to about 8. Such qualitative properties can be exploited in designing a more robust controller. (See [36] for a detailed discussion of these issues in biological mechanisms.)

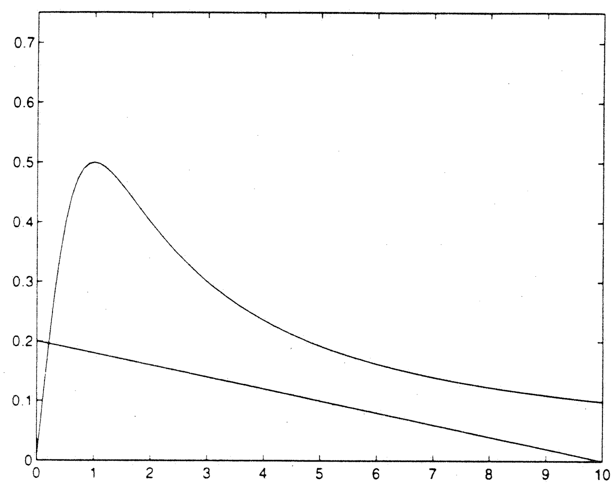

Fig. 4.5. Single solution (when $a = 0.2$ and $b = 10$)

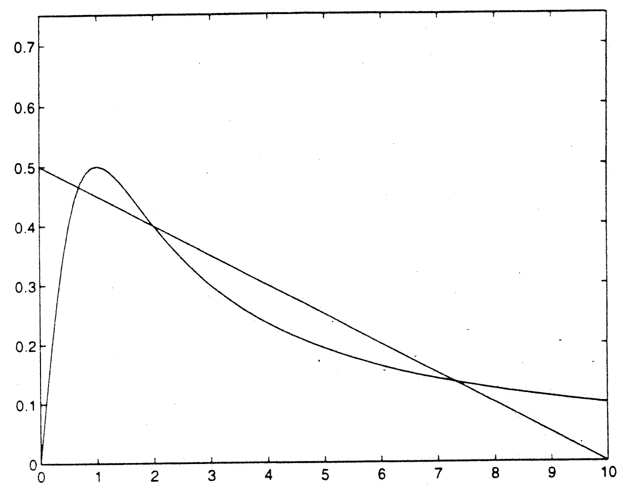

Fig. 4.6. Multiple solutions (when $a = 0.5$ and $b = 10$)

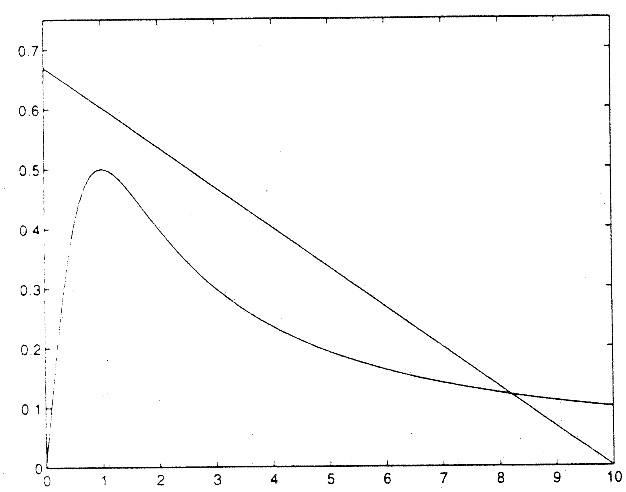

Fig. 4.7. A jump in solutions (when $a = 0.67$ and $b = 10$)

5. Conclusions

In this chapter we have presented an overview of current issues as well as some new directions in multisensor fusion and control. We have described an approach to the design and execution of wide area sensor networks. In addition, we have proposed new techniques for obtaining more robust multisensor fusion systems. However, one very important new area that we have not covered is the ability of multisensor fusion systems to learn during execution (see [43] for an approach to that). These are very exciting times, and we believe that major strides will be made in all these areas in the next few years.

References

[1] Abidi M A, Gonzalez R C 1993 *Data Fusion in Robotics and Machine Intelligence.* Academic Press, Boston, MA

[2] Allen P, Miller A T, Oh P Y, Leibowitz B S 1996 Integration of vision and force sensors for grasping. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 349–356

[3] Bajcsy R, Allen, P 1990 Multisensor integration. In: Shapiro S C (ed) *Encyclopedia of Artificial Intelligence.* Wiley, New York, pp 632–638

[4] Betgé-Brezetz S, Chatila R, Devy M, Fillatreau P, Nashashibi F 1996 Adaptive localization of an autonomous mobile robot in natural environments. In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Las Vegas, NV, pp 77–84

[5] Boissier O, Demazeau Y 1994 MAVI: A multi-agent system for visual integration. In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Las Vegas, NV, pp 731–738

[6] Brooks R R, Iyengar S S 1996 Maximizing multi-sensor system dependability. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 1–8

[7] Cauthorn, J 1997 Load profiling and aggregation using distributed measurements. In: *Proc 1997 DA/DSM DistribuTECH.* San Diego, CA

[8] Cooper S, Durrant-Whyte H 1996 A frequency response method for multisensor high-speed navigation systems. In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Las Vegas, NV, pp 1–8

[9] Dekhil M, Henderson T C 1996 Instrumented sensor systems. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 193–200

[10] Dekhil M, Henderson T C 1997 Instrumented sensor system architecture. Tech Rep UUCS-97-011, University of Utah

[11] Donald B R 1995 On information invariants in robotics. *Artif Intel.* 72:217–304

[12] Durrant-Whyte H 1988 *Integration, Coordination and Control of Multisensor Robot Systems.* Kluwer, Boston, MA

[13] Freund E, Rossmann J 1996 Intuitive control of a multi-robot system by means of projective virtual reality. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 273–280

[14] Garvey T D 1987 A survey of AI approaches to the integration of information. In: *Proc 1987 SPIE Conf Infrared Sensors Sens Fusion*. pp 68–82

[15] Gelb A 1974 *Applied Optimal Estimation*. MIT Press, Cambridge, MA

[16] Griss M L, Kessler R R 1996 Building object-oriented instrument kits. In: *Object Mag*.

[17] Hackett J K, Shah M 1990 Multisensor fusion: A perspective. In: *Proc 1990 IEEE Int Conf Robot Automat*. Cincinnati, OH, pp 1324–1330

[18] Hager G 1990 *Task-directed Sensor Fusion and Planning*. Kluwer, Boston, MA

[19] Henderson T C, Shilcrat E 1984 Logical sensor systems. *J Robot Syst*. 1:169–193

[20] Henderson T C, Weitz E, Hansen C, Mitiche A 1988 Multisensor knowledge systems: interpreting 3D structure. *Int J Robot Res*. 7(6):114–137

[21] Iyengar S S, Prasad L 1995 A general computational framework for distributed sensing and fault-tolerant sensor integration. *IEEE Trans Syst Man Cyber*. 25:643–650

[22] Iyengar S S, Sharma M B, Kashyap R L 1992 Information routing and reliability issues in distributed sensor networks. *IEEE Trans Sign Proc*. 40:3012–3021

[23] Joshi R, Sanderson A 1996 Multisensor fusion and model selection using a minimal representation size framework. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst*. Washington, DC, pp 25–32

[24] Kak A 1987 A production system environment for integrating knowledge with vision data. In: *Proc 1987 AAAI Work Spatial Reason Multisens Fusion*. St. Charles, IL, pp 1–12

[25] Kamberova G, Mandelbaum R, Mintz M 1996 Statistical decision theory for mobile robotics: Theory and application. In: *Proc 1996 IEEE Int Conf Multis Fusion Integr Intel Syst*. Washington, DC, pp 17–24

[26] Korona Z, Kokar M 1996 Model theory based fusion framework with application to multisensor target recognition. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst*. Washington, DC, pp 9–16

[27] Kristensen S, Christensen H I 1996 Decision-theoretic multisensor planning and integration for mobile robot navigation. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst*. Washington, DC, pp 517–524

[28] Luo R, Kay M G 1989 Multisensor integration and fusion in intelligent systems. *IEEE Trans Syst Man Cyber*. 19:901–931

[29] Mann R C 1987 Multisensor integration using concurrent computing. In: *Proc 1987 SPIE Conf Infrared Sensors Sens Fusion*. pp 83–90

[30] Matsuyama T 1994 Belief formation from observation and belief integration using virtual belief space in Dempster-Shafer probability model. In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst*. Las Vegas, NV, pp 379–386

[31] Meystel A 1995 Multiresolutional semiosis. In: *Proc 1996 IEEE Int Symp Intel Contr*. Monterey, CA, pp 13–20

[32] Mitiche A, Aggarwal J K 1986 An overview of multisensor systems. *SPIE Opt Comp*. 2:96–98

[33] Mitiche A, Aggarwal J K 1986 Multisensor integration/fusion through image processing. *Opt Eng*. 25:380–386

[34] Mueller-Planitz C, Kessler R R 1997 Visual threads: The benefits of multithreading in visual programming languages. Tech Rep UUCS-97-012, University of Utah

[35] Murphy R 1996 Adaptive rule of combination for observations over time. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst*. Washington, DC, pp 125–131

[36] Murray J D 1993 *Mathematical Biology*. Springer-Verlag, Berlin, Germany

[37] Mutambara A G O, Durrant-Whyte H F 1994 Modular scalable robot control. In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Las Vegas, NV, pp 121–127

[38] Nakauchi Y, Mori Y 1996 Emergent behavior based sensor fusion for robot navigation system. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 525–532

[39] Rander P W, Narayanan P J, Kanade T 1996 Recovery of dynamic scene structure from multiple image sequences. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 305–312

[40] Rus D, Kabir A, Kotay K D, Soutter M 1996 Guiding distributed manipulation with mobile sensors. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 281–288

[41] Sakaguchi Y, Nakano K 1994 Haptic recognition system with sensory integration and attentional perception In: *Proc 1994 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Las Vegas, NV, pp 288–296

[42] Takahashi E, Nishida K, Toda K, Yamaguchi,Y 1996 CODA: Real-time parallel machine for sensor fusion – Evaluation of processor architecture by simulation. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 836–844

[43] van Dam J W M, Krose B J A, Groen, F C A 1996 Adaptive sensor models. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 705–712

[44] Voyles R M, Morrow J D, Khosla P 1996 Including sensor bias in shape from motion calibration and sensor fusion. In: *Proc 1996 IEEE Int Conf Multisens Fusion Integr Intel Syst.* Washington, DC, pp 93–99

[45] Warrior, J 1997 Smart sensor networks of the future. *Sensors.* 2:40–45

[46] Weller G A, Groen F C A, Hertzberger L O 1990 A sensor processing model incorporating error detection and recovery. In: Henderson T (ed) *Traditional and Non-Traditional Robotic Sensors.* Springer-Verlag, Berlin, Germany, pp 351–363

Lecture Notes in Control and Information Sciences 230

B. Siciliano and K.P. Valavanis (Eds)

Control Problems in Robotics and Automation

IEEE

CSS

Springer