2000 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2000) • Nashville, Tennessee • Oct. 8–11, 2000

Dynamic Dispatching of Coordinated Sensors

Michael D. Naish   Elizabeth A. Croft†   Beno Benhabib

Computer Integrated Manufacturing Laboratory
University of Toronto, 5 King's College Road
Toronto, ON, Canada M5S 3G8
[naish, beno]@mie.utoronto.ca

†Industrial Automation Laboratory
University of British Columbia, 2324 Main Mall
Vancouver, BC, Canada V6T 1Z4
ecroft@mech.ubc.ca

Abstract

Sensory data must be collected in-real time for the majority of autonomous decision making tasks, such as target tracking, surveillance and navigation. The use of multiple sensors may significantly improve the quality and robustness of the data. Given an environment containing a set of mobile sensors, capable of altering their position and orientation, this work addresses the problem of selecting and maneuvering subsets of these sensors for optimal data acquisition in real-time. A heuristic approach to the dispatching problem suitable for on-line implementation is illustrated by a computer-simulated example.

1 Introduction

A variety of autonomous decision making tasks, such as target tracking, surveillance, mobile robot navigation, and robotic soccer, require sensory data collected in real-time. The quality and robustness of the data may be significantly improved through the use of multiple sensors. Specifically, sensor fusion may be used to combine information from multiple sensors into a single representation [1]. Redundant information may be acquired by a group of sensors perceiving the same feature(s) in the environment. By fusing this information, the accuracy of the system is increased. When the features of interest are complex or spatially distributed, complementary sensor groups may be used to perceive features that are imperceptible to individual sensors. Each sensor provides a subset of the required feature space; these feature subsets are combined to obtain the intact feature.

Dynamic dispatching can be effectively utilized to adjust the sensor set on-line to provide the best information possible. This involves both selecting an appropriate subset of sensors to be used in a sensor fusion process and maneuvering all sensors in response to the motion of the object. Namely, the sensors provide information of sufficient quality for the task at hand while ensuring adequate response to object maneuvers (keeping all sensors "in the game"). This is a reactive procedure; therefore, no absolute condition of optimality is imposed. For static sensors, dispatching considers which sensors should be included in the sensor fusion process for during data-acquisition. Dynamic sensor dispatching must, in addition, address the motion of each sensor.

1.1 Dispatching

The dispatching problem has been investigated by a number of researchers in varying fields. Powell discusses and compares a number of different optimization approaches to the problem of dynamic vehicle allocation (DVA) [2]. Psaraftis uses the *rolling horizon* principle for assignment of cargo to ships destined to various ports, [3]. Tentative assignments are made to eligible ships; permanent assignments are made only for those cargoes that fall at the beginning of the rolling horizon (more immediate events are known with greater certainty). The rolling horizon is then shifted to the next time step. A neural network solution for a similar dynamic vehicle dispatching problem (parcel pick-up and delivery or ambulance service) was developed by Potvin *et al.* [4]. Recently, case-based reasoning was employed to determine suitable maneuvers for robotic soccer players [5].

In general, optimization techniques (deterministic and stochastic networks, Markov decision theory, etc.), such as those mentioned above, are not particularly well suited to real-time applications, and neural networks and other learning techniques must be trained for each new situation. An alternate approach is to utilize heuristics. Though not a rigorous treatment of the problem (i.e., stability and optimality cannot be guaranteed), heuristics can provide a tractable, timely, and understandable solution.

A number of heuristic rules, adapted from the dispatching of service vehicles [6], may be suitable for the assignment of sensors to demand points as defined in this paper:

Random sensor rule: Assign a sensor randomly from the set of sensors, regardless of pose with respect to demand point.

Nearest sensor rule: Assign the sensor with the minimum distance to the demand point.

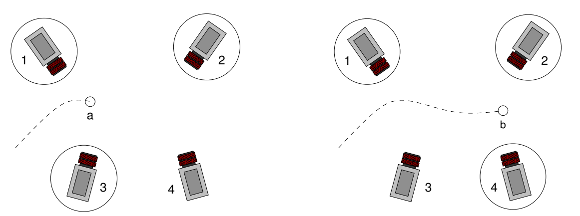
Longest idle sensor rule: Assign the sensor that has remained unassigned for the longest time period.

A fuzzy rule-based system may also be used to trade-off among multiple heuristics, [7], where the heuristics may be refined through the use of genetic programming [8].

In all of the above systems, the assignment of an entity to a particular demand also implies its position. The entity must move to the (fixed) demand point location. As none of these systems consider sensor fusion (or any other form of coordination), each demand point is assigned only one entity.

1.2 Sensor Fusion

In this paper, the primary objective of the sensing-system configuration is to utilize multiple sensors to improve the accuracy and reliability of the sensing-system. Sensors to be used in a sensor fusion process are grouped into *fusion subsets*. A sensor may belong to multiple fusion subsets, as illustrated by Figure 1. A *coordination strategy* specifies the composition of the fusion subsets and is correlated with the time-varying motion of the object. The data provided by each fusion subset is processed by a data fusion methodology. The choice of fusion methodology must satisfy a number of requirements. First, data fusion should require a minimum amount of *a priori* modelling data for characterization of the sensors, objects, and environment. Second, the uncertainty associated with each sensor measurement must be estimated and managed. Finally, the fusion operation should be computationally simple enough that estimates may be provided in real-time, for on-line implementations.



(a) Demand point: a. Fusion subset: 1-2-3.

(b) Demand point: b. Fusion subset: 1-2-4.

Figure 1: Coordination strategy for two demand points using 4 static sensors. Operational sensors are circled.

Specific methodologies suitable for parameter estimation include the least squares estimator (LS) and its variations (WLS, BWLS, MLE, MSE)[1] [9], geometric fusion [10], and the Kalman filter (KF) [11]. The KF is particularly suitable for the on-line processing of measurement data. It may be formulated entirely as scalar equations making it computationally efficient. The basic form of the KF requires that the system be linear; non-linear systems are typically handled with an extended Kalman filter (EKF) [12].

2 Overview of Proposed Method

2.1 Problem Definition

The general dynamic dispatching problem, addressed in this paper, may be stated as: given a set of sensors and a set of time-varying demand points (based on the predicted motion of a maneuvering object), determine a subset of sensors (and their corresponding poses) that will (optimally) sense each demand point and ensure that the remaining sensors are adequately distributed throughout the workspace.

Dispatching does not alter the sensor set, i.e., no sensors are added or removed. A sensor is designated as *operational*, if, at time $t = t^*$, the information it provides is considered in a sensor fusion process. All sensors unused at time $t = t^*$ are designated as *non-operational*. Dynamic dispatching naturally implies continuous pose adjustments for dynamic sensors. This does not imply that the sensor is in continual motion, but rather, that it has the ability to move.

For all dispatching activities, it is assumed that a prediction module provides the system with estimation of the parameterized location, $\mathbf{L}(t)$, and velocity, $\dot{\mathbf{L}}(t)$, of a coordinate frame, F_0, or frames, $\mathbf{F}_0$ (for multiple features), of the moving target [13].

There are two principal strategies implemented by the dispatcher: (1) *coordination strategy* that determines a subset of operational sensors that will be assigned to sense a demand point (the assignment problem); and (2) *surveillance strategy* that positions all sensors, both assigned and unassigned (the positioning problem). The surveillance strategy specifies continuous pose adjustments for dynamic sensors.

2.2 Solution Method

An effective dispatching system requires a number of different types of information, including the following:

Object model: This represents the expected motion of an object through the workspace.

Sensor models: There are two types of sensor models: (1) motion characteristics (range, maximum velocity, etc.)—used to estimate achievable poses within the workspace, and (2) sensing characteristics (accuracy, resolution, field of view, etc.)—used to determine the quality of information that sensor can provide about an object.

Task requirements: These include the maximum sensing accuracy, sensing frequency, size of fusion subset, etc.

An overview of the solution approach proposed herein is illustrated in Figure 2. This approach simplifies the dynamic sensing problem by discretizing the object motion into a number of demand points, P_j. Demand points are predicted at equal time intervals, Δ (constant sampling frequency). (Note that, the demand point frequency is normally much lower than the maximum data acquisition frequency, to allow for corrective sampling during the current interval.) The number of demand points predicted is dependent on the rolling horizon size, which may vary during the task.

A dispatching module is responsible for assigning sensors to the predicted demand points. The assignment of sensors to demand points is based on a rolling horizon, $j\Delta$ long. The

[1]Here, WLS = weighted least squares, BWLS = Bayesian weighted least squares, MLE = maximum likelihood estimate, MSE = minimum squared error.

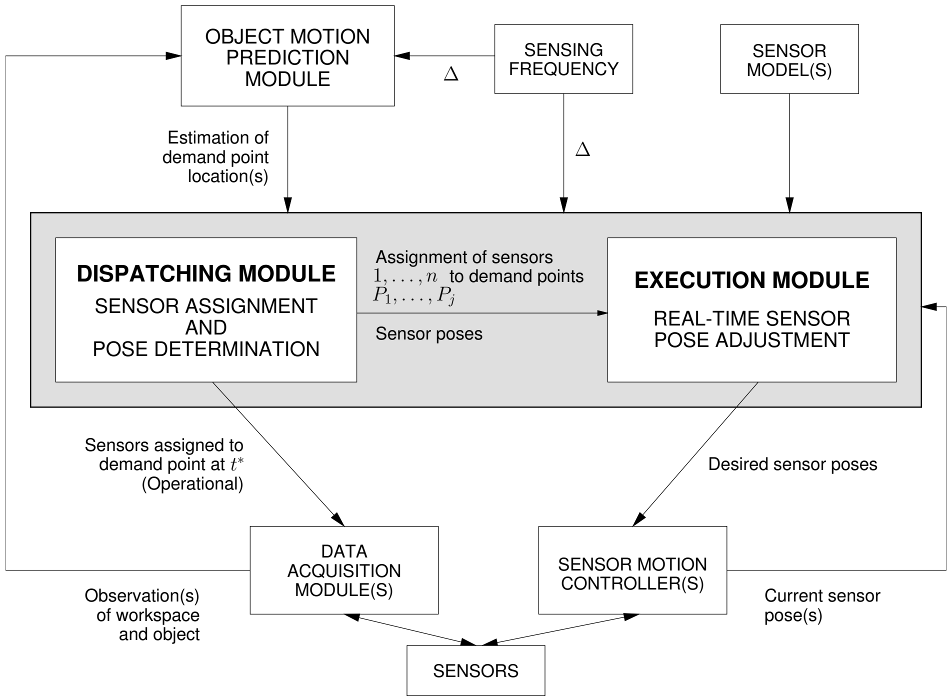


Figure 2: Overview of dynamic dispatching.

assignment of sensors to the current demand point, P_1, once made, is fixed. This determines the coordination strategy. The surveillance strategy is implemented by assigning the remaining sensors to future demand points, $P_2, \ldots, P_j$. This approach aims to ensure that the object may be adequately sensed at all times and that sensors are positioned in anticipation of future sensing demands. Once the current interval has lapsed, all sensors may again be considered for further assignment. Thus, the assignment of sensors to future demand points, $P_2, \ldots, P_j$ is valid only for the duration of the interval.

Integral to the assignment problem is the determination of the "best" pose for each sensor at the end of the interval Δ. Thus, assignment both selects sensors and specifies their pose relative to the demand point. These poses are then passed to an execution module for possible minor modifications.

The execution module is responsible for directing the assigned sensors into the correct poses (relative to the demand points). The initial poses are determined by the dispatching module. As the interval lapses (the time remaining in the interval before the assigned sensors must become operational for the current demand point, $\delta \rightarrow 0$), the prediction of each demand point location is continually updated (and presumably improved). With each update of demand point location, the (desired) pose of each sensor can be adjusted. Note that, these adjustments may be made while the sensor is in motion.

Sensor assignment and relative pose determination implements the *formation strategy* of the sensing-system. Note that, if sensor failure is a concern or obstacles are present in the workspace (that may occlude viewpoints from assigned sensors), the assignment of sensors to a particular demand point may be reassessed more than once over the interval Δ. This would ensure that all k assigned sensors can provide useful data.

3 Visibility Measure

Visibility can be used as a measure of sensor quality and related directly to the measurement uncertainty. Thus, as the sensor pose changes relative to a demand point (object) location, the visibility measure also changes. Ideally, the sensor pose is adjusted such that the visibility measure is maximal.

A visibility measure for a single sensor is considered in this paper as:

$$v_1 = \begin{cases} \dfrac{1}{\|R\|} & \text{if demand point is unoccluded,} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where, $\|R\|$ is the Euclidean norm of the covariance matrix associated with the sensor measurement. Visibility for a configuration of k sensors is defined as:

$$v_k = \frac{1}{\|P\|} \tag{2}$$

where,

$$P = \left[\sum_{i=1}^{k} {R'}_i^{-1} \right]^{-1} \tag{3}$$

and

$$R'_i = \begin{cases} R_i & \text{if demand point is unoccluded,} \\ \emptyset & \text{otherwise.} \end{cases} \quad (4)$$

Namely, the covariance matrix of the ith sensor, R_i, (and hence its visibility) is considered only if the sensor has an unoccluded view of the demand point.

4 Assignment

The dispatching module determines how the sensors will be distributed throughout the workspace by ensuring that each sensor is assigned to a demand point (corresponding to a future object position). If the object trajectory were known *a priori*, it would be possible to obtain a globally optimal solution; however, without complete *a priori* knowledge, an on-line dynamic dispatching system can, at best, only produce locally optimal solutions based on the estimated locations of the current and future demand points. The following sections consider both a locally optimal and heuristic approach (suitable for real-time application) to the dispatching problem.

4.1 Optimal Solution

The optimal approach to sensor assignment considers the assignment of all sensors. This can be accomplished in three stages: First, the number of demand points under consideration is selected (starting from a minimal number and increased, if necessary, until all sensors have been assigned to at least one demand point). Second, combinations of k sensors are selected for each demand point. Finally, the best achievable pose for each sensor with respect to its demand point is determined. From these poses, the visibility of the sensor set is evaluated. The set of sensor combinations and poses that maximizes the visibility measure is selected as the optimal solution. An algorithm that implements this three level search for the locally optimal assignment for each sensor could be as follows:

1. Let $j = \lceil n/k \rceil$, where j indicates the number of demand points to be considered, n is the total number of sensors, and k is the fusion subset size for each demand point.
2. Obtain predictions of (future) demand-point locations for each demand point, at $t = t_{\text{cur}} + \delta$, $t = t_{\text{cur}} + \delta + \Delta, \ldots, t = t_{\text{cur}} + \delta + (j-1)\Delta$ (where, t_{cur} is the current time, δ is the time remaining in the interval before the sensor must be operational for the demand point).
3. Use a combinatorial search technique (e.g., genetic algorithm) to select j combinations of k sensors, from the set of n sensors—one combination for each demand point, $P_1, \ldots, P_j$. The same sensors may belong to multiple subsets. As this is a combinatorial search problem, subsets can be represented as a jn-bit binary string (consisting of 0s and 1s). For example, if $j = 2$ and $n = 6$, a possible string is $\{$`1 0 0 1 1 0|0 1 0 1 0 1`$\}$. This states that Sensors 1, 4, and 5 are assigned to the subset for demand point P_1, and Sensors 2, 4, and 6 are assigned to the subset for demand point P_2. Since the objective of the search is to determine j subsets of size k, a constraint on the search is:

$$\sum_{i=(m-1)n+1}^{mn} s[i] = k, \quad m = 1, \ldots, j, \quad (5)$$

   where $s[i]$ is the ith bit of the string. Namely, the subset assigned to each demand point must consist of exactly k sensors.
4. For each combination, use constrained non-linear optimization (e.g., the flexible tolerance method [14]) to obtain the best achievable pose of each sensor in the combination with respect to the demand point P_j over the time interval $\delta + (j-1)\Delta$ (the time remaining in the interval before the assigned sensors must be operational for the demand point). Choose the combinations (j subsets of k sensors) that provide the optimal visibility:

$$\max v_k, \quad (6)$$

   where, v_k is as defined by Equation (2).
5. If all n sensors have been assigned to a demand point, go to Step 8.
6. Let $j = j + 1$.
7. If $j \leq j_{\max}$ go to Step 2. Here, $j_{\max}$ places an upper limit on number of demand points that may be considered.
8. Starting with P_1, traverse the set of demand points: For each sensor assigned to the demand point, determine if it has already been assigned to another demand point. If not, output demand point and associated desired pose. This will ensure that a single assignment and desired pose is output for each sensor, with higher priority demand points taking precedence (e.g., P_1 has priority over P_2, etc.).

4.2 Heuristic Solution

More suitable for an on-line implementation, the heuristic approach to sensor assignment follows the general principle of the optimal solution; however, instead of optimizing each combination, only one sensor is considered at a time. Starting with demand point P_1, the visibility of each sensor is evaluated for future demand points (maximizing Equation (1)). These visibility measures are then used to rank each sensor with respect to each demand point. The best k sensors are then assigned to their respective demand points. This is repeated until all sensors have been assigned, $j_{\max}$ has been reached, or no time remains in the interval.

A heuristic algorithm for assignment can be as follows:

1. Let $j = 1$, where j is the demand point index.
2. Obtain prediction of demand point location, P_j, for $t = t_{\text{cur}} + \delta + (j-1)\Delta$.
3. Let $i = 1$, where i is the index for the sensors in the set.
4. If $\epsilon \geq \delta$ (where, ϵ is the time it takes to evaluate the visibility of a single sensor) and δ is the time remaining in the interval before the sensor must be operational for the demand point), then go to Step 12.
5. Else, i.e., $\epsilon < \delta$, use the current pose of sensor i and the motion-characteristics model of the sensor to align the sensor axis with P_j and minimize the Euclidean distance between P_j and the sensor frame. This is considered to be the best achievable pose for the sensor with respect to demand point, P_j (over the interval $j\Delta$). From this pose, evaluate the single sensor visibility, v_i.
6. Place visibility measure (with associated pose) of sensor i into an ordered list for demand point j, V_j, ranked in descending order from best to worst.
7. If $i \leq n$, let $i = i+1$, obtain latest prediction of demand point location and go to Step 4.
8. For demand point P_j, assign first k sensors from associated visibility list, V_j. If visibility list is empty, report a failure to assign.
9. If all n sensors have been assigned to a demand point, go to Step 12.
10. Let $j = j + 1$.
11. If $j \leq j_{\max}$ go to Step 2.
12. Starting with P_1, traverse the set of demand points: For each sensor assigned to a demand point, determine if it has already been assigned to another demand point. If not, output demand point and associated desired pose. This will ensure that a single assignment and desired pose is output for each sensor, with earlier demand points taking precedence over later demand points.

5 Execution

The execution module is in control of the sensor's motion. As demand-point estimates are improved (using more recent observations of the object motion), the sensor poses (initially determined by the dispatching module) can also be adjusted by this module, if needed.

The execution module works as follows: First, the module checks whether there is enough time remaining to complete a pose-adjustment iteration ($\epsilon < \delta$). If not, the module waits until $\delta = 0$, at which time a new set of assignments and desired poses will be determined by the dispatching module. Otherwise, if there is sufficient time, the latest predictions of the demand point locations: P_1 at $t = t_{\text{cur}} + \delta$, P_2 at $t = t_{\text{cur}} + \delta + \Delta, \ldots, P_j$ at $t = t_{\text{cur}} + \delta + (j-1)\Delta$, are obtained. Using these predictions, the poses of sensors assigned to each demand point may be adjusted such that the single sensor visibility is maximized over the time interval δ. The goal is to align each sensor axis with the corresponding demand point while minimizing the distance between the demand point and the sensor frame. This process is repeated until the interval has lapsed.

6 An Example Problem

The computational requirements of the optimal solution, discussed in Section 4, makes it unsuitable for real-time applications. As a result, only the heuristic approach to dynamic sensor dispatching is examined below. A simple 2-D example is considered. In this example, range/bearing sensors ($n = 6$) are constrained to rails on the edges of the workspace, but are free to assume any position and orientation along the rail. Thus, each sensor has two degrees of freedom: rotation, α, ($\dot{\alpha}_{\max} = 45$ deg/sec) and horizontal translation, x ($\dot{x}_{\max} = 0.15$ m/sec).

A single dispatching/execution interval is illustrated in Figure 3. This captures the system state at the beginning and end of the interval between 1.6 (grey) and 2.1 (black) seconds (i.e., $\Delta = 0.5$ sec). Both k and $j_{\max}$ are 3. Assignment is based on the poses of the sensors and the three predicted demand point locations at $t = 1.6$, and on the visibility of each sensor for each demand point, Tables 1 and 2. Here, R is computed from the range uncertainty, $\sigma_r^2 = 0.01 + r^2$, and bearing uncertainty, $\sigma_\phi^2 = 0.003 + \phi^2$, where, r is the distance between the sensor and demand point and ϕ is the angle between the sensor axis and the demand point.

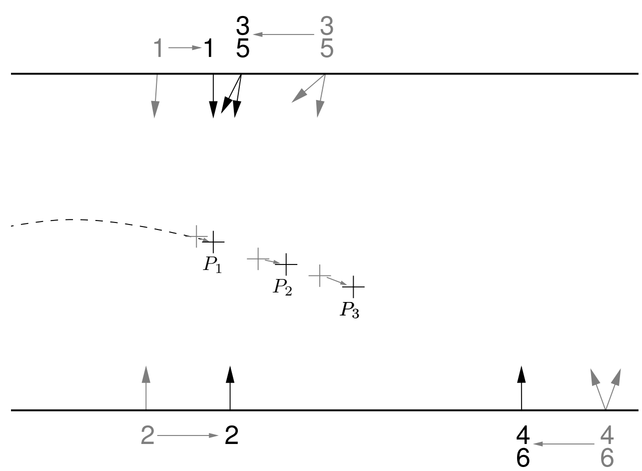


Figure 3: Object and sensor motion during: t = 1.6 to 2.1 sec.

Once assigned, the execution module moves the sensors towards their desired poses. During the interval, new observations (at a frequency of 0.1 sec) are used to update the predicted demand point locations. The desired sensor poses are adjusted accordingly. Thus, at the end of the interval, both the demand point locations and the sensor poses have been altered (shown in black, Figure 3). The final sensor poses for

P_1		P_2		P_3	
Sensor	Visibility	Sensor	Visibility	Sensor	Visibility
1	1.450	2	1.713	2	2.244
5	1.388	1	1.164	4	2.117
3	1.368	5	1.164	6	2.117
2	1.350	3	1.162	1	0.9545
4	0.2840	4	0.7224	3	0.9545
6	0.2145	6	0.6794	5	0.9545

Table 1: Visibility assessment for P_1, $P2$, and P_3.

Sensor	P_j	Initial pose assignment	Final pose at $t = 2.1$
1	1	[0.236, 0.450, 270.0]	[0.252, 0.450, 270.0]
2	2	[0.291, 0.150, 90.0]	in motion
3	1	[0.275, 0.450, 243.1]	[0.275, 0.450, 243.1]
4	3	[0.375, 0.150, 103.8]	in motion
5	1	[0.275, 0.450, 255.1]	[0.275, 0.450, 261.4]
6	3	[0.375, 0.150, 103.8]	in motion

Table 2: Sensor assignment for $t = 1.6$ and pose adjustment.

P_1, at the end of the interval, are given in Column 4 of Table 2. Note that, if the object motion predictions were accurate, the desired sensor poses determined by the dispatcher would be unchanged over the interval. Dispatching improved the fused visibility of the object for P_1 from 3.275 to 4.103.

At $t = 2.1$ sec, the dispatching/execution cycle is repeated: the demand point locations are predicted for the next interval and the sensors are reassigned accordingly. Sensors 1, 2, and 5 are assigned to P_2 and Sensors 2, 4, and 6 are assigned to both P_3 and P_4. Sensor 3, without assignment, remains stationary for the interval.

7 Conclusions

A method for maximizing the effectiveness of a set of sensors is presented in this paper. The overall goal of the method is to position sensors in response to changing demands. This is accomplished using two modules, one for dispatching and one for execution, that together implement the desired coordination and surveillance strategies. Sensors are evaluated based on the quality of information that each can provide for specified object locations. From this, a group of sensors (for use in a sensor fusion context) may then be assigned to a particular sensing demand. In addition, the sensors that are not required for the most imminent demand, are assigned to future predicted demands. This ensures that as many sensors as possible are maneuvered in anticipation of upcoming requirements rather than remaining idle or moving randomly.

The sensor configuration is adjusted according to a continual reevaluation of the capabilities of each sensor and the sensing requirements. This is a reactive procedure, executed on-line; therefore, no absolute condition of optimality is imposed. In fact, the ability of the dispatching and execution modules to effectively adjust the sensor poses is very dependent on the maneuverability of the object and sensors, the initial poses of the sensors, the number of sensors used, and the size of each fusion subset. Variations in each of these parameters, in addition to the accuracy of the object motion prediction, can drastically affect the overall performance of the system.

Acknowledgements

The financial support of the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.

References

[1] R. C. Luo and M. G. Kay, "Multisensor integration and fusion for intelligent machines and systems," in *Data Fusion in Robotics and Machine Intelligence*, pp. 7–136, San Diego, CA: Academic Press, 1992.

[2] W. B. Powell, "A comparative review of alternative algorithms for the dynamic vehicle allocation problem," in *Vehicle Routing: Methods and Studies*, pp. 249–291, Amsterdam: North-Holland, 1988.

[3] H. N. Psaraftis, "Dynamic vehicle routing problems," in *Vehicle Routing: Methods and Studies*, pp. 223–248, Amsterdam: North-Holland, 1988.

[4] J.-Y. Potvin and Y. Shen, "A neural network approach to the vehicle dispatching problem," in *Proc. IEEE Int. Joint Conf. on Neural Networks*, pp. 1230–1235, Nov. 18-21, 1991.

[5] J. Wendler, P. Gugenberger, and M. Lenz, "CBR for dynamic situation assessment in an agent-oriented setting," in *Proc. AAAI-98 Workshop on Case-Based Reasoning Integrations*, July 27, 1998.

[6] N. Viswanadham and Y. Narahari, *Performance Modeling of Automated Manufacturing Systems*, pp. 90–94. Englewood Cliffs: Prentice Hall, 1992.

[7] M. Shrivastava, P. K. Chande, and A. S. Monga, "Taxi dispatch: A fuzzy rule approach," in *Proc. IEEE Conf. on Intelligent Transportation Systems*, pp. 978–982, Nov. 9-12, 1997.

[8] I. Benyahia and J.-Y. Potvin, "Decision support for vehicle dispatching using genetic programming," *IEEE Trans. on Sys., Man, and Cybernetics*, vol. 28, pp. 306–314, May 1998.

[9] D. L. Hall, *Mathematical Techniques in Multisensor Data Fusion*. Boston, MA: Artech House, 1992.

[10] Y. Nakamura and Y. Xu, "Geometrical fusion method for multisensor robotic systems," in *Multisensor Integration and Fusion for Intelligent Machines and Systems*, pp. 241–259, Norwood, NJ: Ablex, 1995.

[11] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, 1993.

[12] T. J. Broida, "Kinematic and statistical models for data fusion using Kalman filtering," in *Data Fusion in Robotics and Machine Intelligence*, pp. 267–309, San Diego, CA: Academic Press, 1992.

[13] D. He, D. Hujic, J. K. Mills, and B. Benhabib, "Moving-object recognition using premarking and active vision," in *Proc. IEEE International Conf. on Robotics and Automation*, pp. 1980–1985, 1996.

[14] D. M. Himmelblau, *Applied Nonlinear Programming*. New York: McGraw-Hill, 1972.