A Comparison of Decision Making Criteria and Optimization Methods for Active Robotic Sensing

L. Mihaylova, T. Lefebvre, H. Bruyninckx, K. Gadeyne and J. De Schutter

Katholieke Universiteit Leuven, Celestijnenlaan 300B, B-3001 Heverlee Belgium,
E-mail: *Lyudmila.Mihaylova@mech.kuleuven.ac.be*

Abstract. This work presents a comparison of decision making criteria and optimization methods for active sensing in robotics. *Active sensing* incorporates the following aspects: (*i*) where to position sensors, and (*ii*) how to make decisions for next actions, in order to maximize information gain and minimize costs. We concentrate on the second aspect: "Where should the robot move at the next time step?". Pros and cons of the most often used statistical decision making strategies are discussed. Simulation results from a new multisine approach for active sensing of a nonholonomic mobile robot are given.

1 Introduction

One of the features of robot intelligence is to deal robustly with uncertainties. This is only possible when the robot is equipped with sensors, e.g., contact sensors, force sensors, distance sensors, cameras, encoders, gyroscopes. To perform a task, the robot first needs to know: "Where am I now ?". After that the robot needs to decide "What to do next ?", weighting future information gain and costs. The latter decision making process is called *active sensing*. Distinction is made sometimes between active sensing and active localization. "Active localization" refers to robot motion decisions (e.g. velocity inputs), "active sensing" to sensing decisions (e.g. when a robot is allowed to use only one sensor at a time). In this paper we refer to both strategies as "active sensing". Choosing actions requires to trade off the immediate with the long-term effects: the robot should take both actions to bring itself closer to its *task completion* (e.g. reaching a goal position within a certain tolerance) and actions for the purpose of *gathering information*, such as searching for a landmark, surrounding obstacles, reading signs in a room, in order to keep the uncertainty small enough at each time instant and assure a good task execution. Typical tasks where active sensing is useful are performed in less structured environments. The uncertainties are so important that they influence the task execution: industrial robot tasks in which the robot is uncertain about the configuration (positions and orientation) of its tool and work pieces [1]; mobile robot navigation in a known map (indoor and outdoor) [2, 3] where starting from an uncertain initial configuration the robot has to move to a desired goal configuration within a preset time; vision applications with active selection of camera parameters such as focal length and viewing angle to improve the object recognition procedures [4, 5]; reinforcement learning [6]: the robot needs to choose a balance between its localization (*exploiting*) and the new information it can gather about the environment (*exploring*).

Estimation, control and active sensing. Next to an active sensing module, intelligent robots should also include an estimator and a controller:

- *Estimation.* To overcome the uncertainty in the robot and environment models, as well as the sensor data, estimation techniques [7, 8] compute the system state after fusing the data in an optimal way.
- *Control.* Knowing the desired task, the controller is charged with following the task execution as closely as possible. Motion execution can be achieved either by feedforward, feedback control or a combination of both [9].
- *Active sensing* is the process of determining the inputs by optimizing a function of costs and utilities. These inputs are then sent to the controller.

Active sensing is challenging for various reasons: (*i*) The robot and sensor models are *nonlinear*. Some methods linearize these models, but many nonlinear problems cannot be treated this way and impose the necessity to develop special techniques for action generation. (*ii*) The task solution depends on an *optimality criterion* which is a *multi-objective* function weighting the information gain and some costs. It is related to the *computational load* especially important for *on-line* task execution. (*iii*) *Uncertainties* in the robot and environment models, the sensor data need to be dealt with. (*iv*) Often measurements do not supply information about all variables, i.e. the system is *partially observable*.

The remainder of the paper is organized as follows. In Section 2, the active sensing problem is described. The most often used decision making criteria are compared and results for active sensing of a nonholonomic mobile robot are presented. Section 3 gives the main groups of optimization algorithms for active sensing. Section 4 terminates with the conclusions.

2 Active sensing : problem formulation

Active sensing can be considered as a trajectory generation for a *stochastic dynamic* system described by the model

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_k) \tag{1}$$

$$\boldsymbol{z}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{s}_{k+1}, \boldsymbol{\xi}_{k+1}) \tag{2}$$

where $\boldsymbol{x}$ is the system state vector, $\boldsymbol{f}$ and $\boldsymbol{h}$ nonlinear system and measurement functions, $\boldsymbol{z}$ is the measurement vector, $\boldsymbol{\eta}$ and $\boldsymbol{\xi}$ are respectively system and measurement noises. $\boldsymbol{u}$ stands for the input vector of the state function, $\boldsymbol{s}$ stands for a sensor parameter vector as input of the measurement function (an example is the focal length of a camera). The subscript k denotes discrete time. The system's states and measurements are influenced by the inputs $\boldsymbol{u}$ and $\boldsymbol{s}$. Further, we make no distinction and denote both inputs to the system with $\boldsymbol{a}$ (actions). Conventional systems consisting only of control and estimation components assume that these inputs are given and known. Intelligent systems should be able to adapt the inputs in a way to get the "best" estimates and in the meanwhile to perform the *active sensing* task "as good as possible".

So, an appropriate *multi-objective performance criterion* (often called *value function*) is needed to quantify for each sequence of actions $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$ (also called *policy*) both the information gain and the gain in task execution:

$$J = \min_{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N} \{\sum_j \alpha_j \mathcal{U}_j + \sum_l \beta_l \mathcal{C}_l\} \tag{3}$$

This criterion is composed by a weighted sum of *rewards*: (*i*) j terms $\mathcal{U}_j$ characterizing the minimization of *expected uncertainties* (maximization of *expected information extraction*) and (*ii*) l terms $\mathcal{C}_l$ specifying other *expected costs and utilities*, e.g. travel distance, time, energy, distances to obstacles. Both $\mathcal{U}_j$ and $\mathcal{C}_k$ are function of the policy $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$. The weighting coefficients α_j and β_l give different impact to the two parts, and are arbitrarily chosen by the designer. When the state at the goal configuration fully determines the rewards, the terms $\mathcal{U}_j$ and $\mathcal{C}_l$ are computed based on this state only. When attention is paid to both the goal configuration and the intermediate time evolution, the terms $\mathcal{U}_j$ and $\mathcal{C}_l$ are a function of the $\mathcal{U}_{j,k}$ and $\mathcal{C}_{l,k}$ at different time steps k. Criterion (3) is to be minimized with respect to the sequence of actions under *constraints*

$$\boldsymbol{c}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_N) \leq \boldsymbol{c}_{thr}. \tag{4}$$

$\boldsymbol{c}$ is a vector of physical variables that can not exceed some threshold values $\boldsymbol{c}_{thr}$. The thresholds express for instance maximal allowed velocities and acceleration, maximal steering angle, minimum distance to obstacles, etc.

2.1 Action sequence

The description of the sequence of actions $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$ can be done in different ways and has a major impact on the optimization problem that needs to be solved afterwards (Section 3).

- The actions can be described as lying on a reference trajectory plus a *parameterized* deviation of it (e.g. by a finite sine/cosine series, or by an elastic band or elastic strip formulation, [9, 10]). In this way, the optimization problem is reduced to a finite-dimensional optimization problem on the parameters.
- The most general way to present the policy is a sequence of freely chosen actions, not restricted to a certain form of trajectory. Constraints, such as maximal acceleration and maximal velocity, can be added to produce executable trajectories. This active sensing problem is called a *Markov Decision Process* (MDP) for systems with fully observable states and *Partially Observable Markov Decision Process* (POMDP) for systems where measurements do not fully observe the states or for systems with measurement noise.

2.2 Performance criteria related to uncertainty

The terms $\mathcal{U}_j$ represent (*i*) the expected uncertainty of the system about its state; or (*ii*) this uncertainty compared to the accuracy needed for the task completion. In a Bayesian framework, the characterization of the uncertainty of the estimate is based on a scalar loss function of its probability density function. Since no scalar function can capture all aspects of a matrix, no function suits the needs of every experiment. Common used functions are:

- ***based on the covariance matrix*:** The covariance matrix $\boldsymbol{P}$ of the probability distribution of state $\boldsymbol{x}$ is a measure for the uncertainty on the estimate. Minimizing $\boldsymbol{P}$ corresponds to minimizing the uncertainty. Active sensing is looking for actions which minimize the *posterior* covariance matrix $\boldsymbol{P} = \boldsymbol{P}_{post}$ or the inverse of the *Fisher information matrix* $\boldsymbol{I}$ [11] which describes the posterior covariance matrix of an efficient estimator $\boldsymbol{P} = \boldsymbol{I}^{-1}$. Several scalar functions of $\boldsymbol{P}$ can be applied [12] :
  - *D-optimal design*: minimizes the matrix determinant, $det(\boldsymbol{P})$, or the logarithm of it, $log(det(\boldsymbol{P}))$. The *minimum* is *invariant* to any transformation of the state vector $\boldsymbol{x}$ with a non-singular Jacobian such as scaling. Unfortunately, this measure does not allow to verify task completion: the determinant of the matrix being smaller than a certain value does not impose any of the covariances of the state variables to be smaller than their toleranced value.
  - *A-optimal design*: minimizes the trace $tr(\boldsymbol{P})$. Unlike D-optimal design, A-optimal design does not have the invariance property. The measure does not even make sense physically if the target states have inconsistent units. On the other hand, this measure allows to verify task completion.
  - *L-optimal design*: minimizes the weighted trace $tr(\boldsymbol{W}\boldsymbol{P})$. A proper choice of the weighting matrix $\boldsymbol{W}$ can render the L-optimal design criterion *invariant* to transformations of the variables $\boldsymbol{x}$ with a non-singular Jacobian: $\boldsymbol{W}$ has units and is also transformed. A special case of L-optimal design is the tolerance-weighted L-optimal design [1], which proposes a natural choice of $\boldsymbol{W}$ depending on the desired standard deviations (tolerances) at task completion. The value of this scalar function has a *direct relation to the task completion*.
  - *E-optimal design*: minimizes the maximum eigenvalue $\lambda_{max}(\boldsymbol{P})$. Like A-optimal design, this is not invariant to transformations of $\boldsymbol{x}$, nor does the measure makes sense physically if the target states have inconsistent units, but the measure allows to verify task completion.
- ***based on the probability density function***: Entropy [13] is a measure of the uncertainty of a state estimate containing more information about the probability distribution than the covariance matrix, at the expense of more computational costs. The entropy based performance criteria are:
  - the *entropy* of the posterior distribution: $E[-\log p_{post}(\boldsymbol{x})]$. $E[.]$ indicates the expected value.
  - the *change in entropy* between two distributions $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$: $E[-\log p_2(\boldsymbol{x})] - E[-\log p_1(\boldsymbol{x})]$. For active sensing, $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ can be the prior and posterior or the posterior and the goal distribution.
  - the *Kullback-Leibler distance* or *relative entropy* [14] is a measure for the goodness of fit or closeness of two distributions: $E[\log \frac{p_2(\boldsymbol{x})}{p_1(\boldsymbol{x})}]$. The

expected value is calculated with respect to $p_2(\boldsymbol{x})$. The relative entropy and the change in the entropy are *different* measures. The *change in entropy* only quantifies how much the form of the probability distributions changes whereas the *relative entropy* also represents a measure of how much the distribution has moved. If $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ are the same distributions, translated by different mean values, the change in entropy is zero, while the relative entropy is not.

Example. Distance and orientation sensing of a mobile robot to known beacons is considered. The sequence of motions of a nonholonomic wheeled mobile robot (WMR) [15], moving from a starting to a goal configuration, is restricted to a *parameterized trajectory.* The optimal trajectory is searched in the class $\mathcal{Q} = \mathcal{Q}(\boldsymbol{p}), \boldsymbol{p} \in \mathcal{P}$, of harmonic functions, where $\boldsymbol{p}$ is a vector of parameters obeying to preset physical constraints. With N the number of functions, the new (modified) robot trajectory is generated on the basis of a reference one by the lateral deviation l_k (*lateral* is called the orthogonal robot motion deviation from a straight line reference trajectory in y direction) as a linear superposition

$$l_k = \sum_{i=1}^{N} A_i sin(i\pi \frac{s_{r,k}}{s_{r,total}}), \tag{5}$$

of sinusoids, with constant amplitudes A_i, $s_{r,k}$ is the path length up to instant k, $s_{r,total}$ is the total path length, and r refers to the reference trajectory. In this formulation active sensing is a *global optimization problem* (on the whole robot trajectory) with a criterion to be minimized

$$J = \min_{A_{i,k}} \{\alpha_1 \mathcal{U} + \alpha_2 \mathcal{C}\} \tag{6}$$

under *constraints* (for the robot velocity, steering and orientation angles). α_1 and α_2 are dimensionless positive weighting coefficients. Here $\mathcal{U}$ is in the form

$$\mathcal{U} = tr(\boldsymbol{W}\boldsymbol{P}), \tag{7}$$

where $\boldsymbol{P}$ is the covariance matrix of the estimated states (at the goal configuration), computed by an Unscented Kalman filter [16] and $\boldsymbol{W}$ is a weighting matrix). The cost term $\mathcal{C}$ is assumed to be the relative time $\mathcal{C} = t_{total}/t_{r,total}$, where t_{total} is the total time for reaching the goal configuration on the modified trajectory, $t_{r,total}$ the respective time over the reference trajectory. The weighting matrix $\boldsymbol{W}$ represents a product of a normalizing matrix $\boldsymbol{N}$, and a scaling matrix $\boldsymbol{M}$, $\boldsymbol{W} = \boldsymbol{M}\ \boldsymbol{N}$. The matrix $\boldsymbol{N} = diag\{1/\sigma_1^2,\ 1/\sigma_2^2, \ldots, \sigma_n^2\}$. σ_i, $i = 1, \ldots, n$, are assumed here to be the standard deviations at the goal configuration on the reference trajectory. Depending on the task, they could be chosen otherwise. The scaling matrix $\boldsymbol{M}$ here is the identity matrix. Simulation results obtained both with (7), and with the averaged criterion $\mathcal{U}_a = \frac{1}{k_b - k_a} \sum_{k=k_a}^{k_b} tr(\boldsymbol{W_k}\boldsymbol{P_k})$ with optimization over the interval $[k_a, k_b] = [30sec, 100sec]$ are given on Figs. 1, 2. The modified trajectory, generated with different number of sinusoids N (in accordance with (5)), and the reference trajectory are plotted together with the

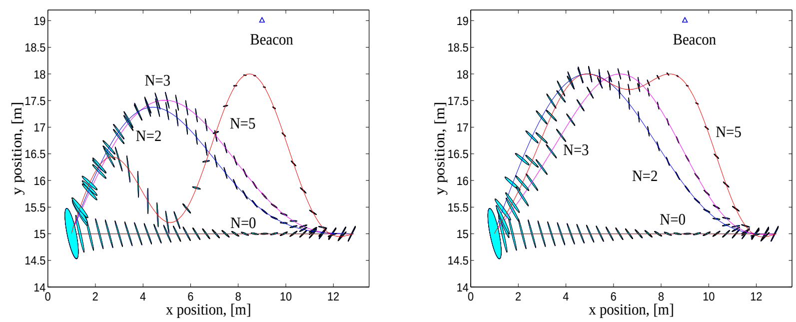

Fig. 1. Trajectories, generated with : (a) $\mathcal{U} = tr(\boldsymbol{WP})$ (b) the averaged criterion $\mathcal{U}_a = \frac{1}{k_b - k_a} \sum_{k=k_a}^{k_b} tr(\boldsymbol{W_k P_k})$

uncertainty ellipses Figs. 1,2. As it is seen from Figs. 1,2 the most accurate results at the goal configuration for $\mathcal{U}$ and J are obtained with $N = 5$ sinusoids. Better accuracy is provided with bigger N, at the cost of increased computational load. Through active sensing the robot is approaching to the beacons (Fig. 1), that is a distinction from a movement over a reference trajectory. Faster increase of the information at the beginning of the modified trajectories and higher accuracy, is obtained than those on the straight-line. From other side, trajectories generated by the averaged criterion $\mathcal{U}_a$ are characterized with better general performance then those generated with (7) (Fig. 2).

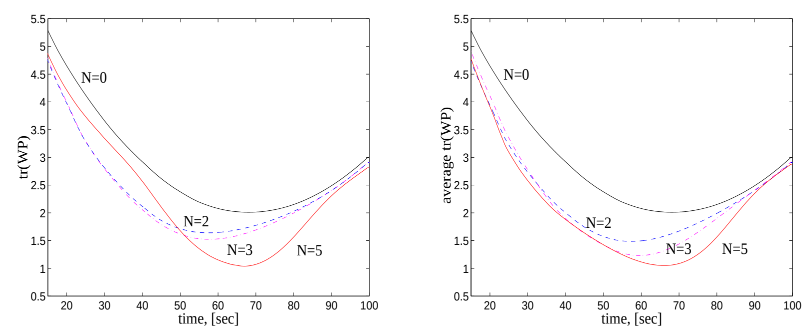

Fig. 2. Performance criteria: (a) $\mathcal{U} = tr(\boldsymbol{WP})$, (b) $\mathcal{U}_a = \frac{1}{k_b - k_a} \sum_{k=k_a}^{k_b} tr(\boldsymbol{W_k P_k})$

3 Optimization algorithms for active sensing

Active sensing corresponds to a constraint optimization of J with respect to the policy $\boldsymbol{a}_1, \ldots \boldsymbol{a}_N$. Depending on the robot task, sensors and uncertainties, different *constraint optimization problems* arise:

- If the sequence of actions $\boldsymbol{a}_1, \ldots \boldsymbol{a}_N$ is restricted to a *parameterized trajectory*, the optimization can have different forms: linear programming, constrained nonlinear least squares methods, convex optimization [17]. Example is the dynamical robot identification [18].
- If the sequence of actions $\boldsymbol{a}_1, \ldots \boldsymbol{a}_N$ is *not restricted to a parameterized trajectory*, then the (PO)MDP optimization problem has a different structure. This could be a finite-horizon, i.e. over a fixed finite number of time steps (N is finite), or an infinite-horizon problem ($N = \infty$). For every state it is rather straightforward to know the immediate reward being associated to every action (1 step policy). The goal however is to find the policy that maximizes the reward over a long term (N steps). Different optimization procedures exist for this kind of problems, examples are:
  - *Value iteration:* due to the sequential structure of the problem, the optimization can be performed as subsequent solution of problems with only 1 (of the N) variables a_i. The *value iteration* algorithm, a *dynamic programming* algorithm, calculate recursively the optimal value function and policy [19] for finite and infinite horizon problems.
  - *Policy iteration:* an iterative technique similar to dynamic programming, is introduced by Howard [20] for infinite horizon systems.
  - *Linear programming:* an infinite horizon problem can be represented and solved as a linear program [21].
  - *State based search methods* represent the system as a graph whose nodes correspond to states and can handle finite and infinite horizon problems [22]. *Tree search* algorithms search for the optimal path in the graph.

  Unfortunately, exact solutions can only be found for (PO)MDPs with a small number of (discretized) states. For larger problems *approximate solutions* are needed [22, 23].

4 Conclusions

This paper addresses the main issues of active sensing in robotics. Multi-objective criteria are used to determine if the result of an action is better than the result of another action. These criteria are composed of two terms: a term characterizing the uncertainty minimization (maximization of information extraction) and a term representing other utilities or costs, such as traveled path or total time. The features of the basic criteria for uncertainty minimization and the optimization procedures are outlined. Simulation results for active sensing of a wheeled mobile robot with a parameterized sequence of actions are presented.

Acknowledgments. Herman Bruyninckx and Tine Lefebvre are, respectively, Postdoctoral and Doctoral Fellows of the Fund for Scientific Research-Flanders (F.W.O–Vlaanderen) in Belgium. Lyudmila Mihaylova is a Postdoctoral Fellow at Katholieke Universiteit Leuven, on leave from the Bulgarian Academy of Sciences. Financial support by the Center of Excellence BIS21 grant ICA1-2000-70016 and the K. U. Leuven's Concerted Research Action GOA/99/04 are gratefully acknowledged.

References

1. J. D. Geeter, J. De Schutter, H. Bruyninckx, H. V. Brussel, and M. Decrton, “Tolerance-weighted L-optimal experiment design: a new approach to task-directed sensing,” *Advanced Robotics*, vol. 13, no. 4, pp. 401–416, 1999.
2. N. Roy, W. Burgard, D. Fox, and S. Thrun, “Coastal navigation - mobile robot navigation with uncertainty in dynamic environments,” in *Proc. of ICRA*, 1999.
3. A. Cassandra, L. Kaelbling, and J. Kurien, “Acting under uncertainty: Discrete Bayesian models for mobile robot navigation,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1996.
4. G. N. DeSouza and A. Kak, “Vision for mobile robot navigation: A survey,” *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 24, no. 2, pp. 237–267, 2002.
5. J. Denzler and C. Brown, “Information theoretic sensor data selection for active object recognition and state estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 145–157, February 2002.
6. R. Sutton and A. Barto, *Reinforcement Learning, An introduction.* MIT, 1998.
7. M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. on Signal Proc.*, vol. 50, no. 2, pp. 174–188, 2002.
8. Y. Bar-Shalom and X. Li, *Estimation and Tracking: Principles, Techniques and Software.* Artech House, 1993.
9. J.-P. Laumond, *Robot Motion Planning and Control.* Guidelines in Nonholonomic Motion Planning for Mobile Robots, by J.-P, Laumond, S. Sekhavat, and F. Lamiraux, available at http://www.laas.fr/~jpl/book.html: Springer-Verlag, 1998.
10. O. Brock and O. Khatib, “Elastic strips: A framework for integrated planning and execution,” in *Proc. of 1999 Int. Symp. of Experim. Robotics*, pp. 245–254, 1999.
11. R. Fisher, “On the mathematical foundations of theoretical statistics,” *Phylosophical Trans. of the Royal Society of London, Series A*, vol. 222, pp. 309–368, 1922.
12. V. Fedorov, *Theory of optimal experiments.* Academic press, New York ed., 1972.
13. C. Shannon, “A mathematical theory of communication, I and II,” *The Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, July and October 1948.
14. S. Kullback, “On information and sufficiency,” *Annals of mathematical Statistics*, vol. 22, pp. 79–86, 1951.
15. L. Mihaylova, J. De Schutter, and H. Bruyninckx, “A multisine approach for trajectory optimization based on information gain,” in *Proc. of IROS Conf.*, 2002.
16. S. Julier, J. Uhlman, and H. Durrant-Whyte, “A new method for the transformation of means and covariances in filters and estimators,” *IEEE Trans. on AC*, vol. 45, no. 3, pp. 477–482, 2000.
17. NEOS, “Argonne national laboratory and northwestern university, optimization technology center,” *http://www-fp.mcs.anl.gov/otc/Guide/.*
18. J. Swevers, C. Ganseman, D. Tukel, J. De Schutter, and H. V. Brussel, “Optimal robot excitation and identification,” *IEEE Trans. on AC*, vol. 13, no. 5, pp. 730–740, 1997.
19. R. Bellman, *Dynamic Programming.* New Jersey: Princeton Univ. Press, 1957.
20. R. A. Howard, *Dynamic Programming and Markov Processes.* MIT Press, 1960.
21. P. Schweitzer and A. Seidmann, “Generalized polynomial approximations in Markovian decision processes,” *J. of MA and Appl.*, vol. 110, pp. 568–582, 1985.
22. C. Boutilier, T. Dean, and S. Hanks, “Decision-theoretic planning: Structural assumptions and computational leverage,” *J. of AI Research*, vol. 11, pp. 1–94, 1999.
23. W. S. Lovenjoy, “A survey of algorithmic methods for partially observed Markov decision processes,” *Annals of Operations Research*, vol. 18, pp. 47–66, 1991.