Adaptive Outcome Selection for Planning with Reduced Models

Sandhya Saisubramanian and Shlomo Zilbertsein

***Abstract*— Reduced models allow autonomous robots to cope with the complexity of planning in stochastic environments by simplifying the model and reducing its accuracy. The solution quality of a reduced model depends on its fidelity. We present *0/1 reduced model* that selectively improves model fidelity in certain states by switching between using a simplified deterministic model and the full model, without significantly compromising the run time gains. We measure the reduction impact for a reduced model based on the values of the ignored outcomes and use this as a heuristic for outcome selection. Finally, we present empirical results of our approach on three different domains, including an electric vehicle charging problem using real-world data from a university campus.**

I. INTRODUCTION

Autonomous robots are often faced with tasks that require generating plans quickly to navigate between two points. Uncertainty in action outcomes, which is a characteristic of many real-world problems, increases the complexity of path planning. These problems can be conveniently modeled as Stochastic Shortest Path (SSP) problems [1], which generalizes finite and infinite-horizon Markov decision processes (MDPs). Since the robots often operate in resource-constrained settings, it is common to plan using reduced (simplified) models of the world that trade solution quality for computational gains [2]. We consider reduced models in which the number of outcomes per action in each state is reduced relative to the original model.

Reduced models simplify the problem and accelerate planning by partially or completely ignoring uncertainty, thereby reducing the set of reachable states for the planner [3], [4]. An example of this is determinization, which simplifies the problem by associating one deterministic outcome for each action, instead of multiple probabilistic outcomes [5], [6]. The resulting deterministic model can be efficiently solved using algorithms such as A* [7]. The possible action outcomes considered in the reduced model determine the model fidelity and hence different outcome selection techniques result in reduced models with varying fidelity.

For example, consider a motivating problem in which a robot must navigate through the corridors in a building. During navigation, it may encounter congested regions with static obstacles or dynamically moving humans (Fig. 2(a)). Reliable navigation around these obstacles and one that is able to obviate conflicts with human traffic can be achieved by reasoning about the robot's drift and probabilistic predictions of the human trajectories [8]. Since the robot needs to handle such situations effectively, we refer to such states as "critical" or "risky" states. Executing sub-optimal actions in these risky states may significantly affect the expected cost of reaching a goal. Hence we introduce an added penalty to account for such risks. While studies of the risks associated with AI systems are attracting growing interest [9]–[12], research of the risks that result from simplifying the planning model has been lacking.

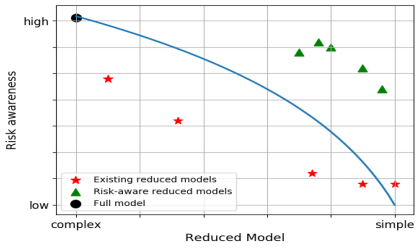


Fig. 1. An example illustrating the trade-off between model simplicity and risk awareness. Risk awareness decreases as the model is simplified using uniform outcome selection principles (indicated by the blue trend line). The points denote reduced models formed with different reduction techniques.

A *risk-aware* reduced model accounts for the possibility of encountering such risky states during plan execution, thus resulting in improved solution quality. A simple reduced model (such as determinization) accelerates planning but may not account for the probabilities of encountering obstacles and conflicts with human trajectories. Conversely, planning with the full probabilistic model is computationally expensive. Hence, the key question we address in this work is *how to create reduced models that balance the trade-off between model simplicity and risk awareness* (Fig. 1).

Intuitively, the trade-off between model simplicity and risk awareness can be optimized by identifying when to use a simple model and when to use a more informed model. For a robot navigating in a building, a plan generated by the robot using a simple reduced model may work well when the robot is moving through an uncluttered region, but a more informative reduced model or the full model is required to reliably navigate through obstacles and humans [2], [4]. The existing reduced model techniques are either incapable of handling such variations as they employ a uniform (non-adaptive) outcome selection approach to all state-action pairs [3], [5], [6], or perform model switching only when no feasible solution is found with a lower fidelity model [4]. This limits the scope of risks they can represent, often resulting in overly optimistic planning model and sub-optimal solutions.

We present planning using $0/1$ reduced models ($0/1$ RM), that enables formulating reduced models with different levels of details by switching between using a deterministic

Support for this work was provided in part by the U.S. National Science Foundation grants IIS-1524797 and IIS-1724101.

College of Information and Computer Sciences, University of Massachusetts Amherst, MA 01003, USA. {saisubramanian, shlomo}@cs.umass.edu

model and the full model. We consider a factored state representation, which allows us to characterize risks in terms of the state features. Precise identification of states where risk awareness is to be improved is particularly complex and generally infeasible without solving the full model. Therefore, we start with a simple reduced model and query an oracle (human) for features indicating the risks in the system. The oracle *guides* the planning process by providing features that indicate risk, which is more realistic since it is relatively easy for humans to identify such features instead of meticulously marking states as risky. Querying an oracle once per domain may be sufficient if the problem instances are similar or share a structure.

To identify states where model fidelity is to be improved, a *reduction impact* is estimated automatically based on the provided features, by generating and solving sample trajectories. The reduction impact measures how optimistic the resulting reduced model would be, with respect to risks, thus offering a heuristic for choosing the outcome selection principles. In states where the reduction impact is high, more informative outcome selection principles are employed. Finally, we evaluate our approach in three domains in simulation. The results demonstrate that our approach efficiently balances the trade-off between risk awareness and model simplicity.

Section III introduces planning using $0/1$ reduced models. Section IV describes the estimation of reduction impact and how it acts as a heuristic for outcome selection. Empirical evaluation of our approach is presented in Section V.

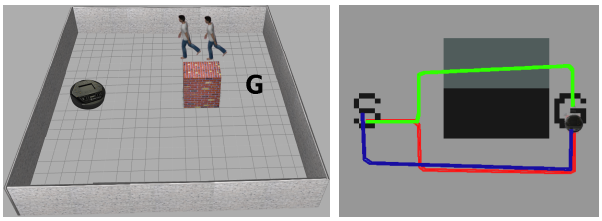
II. Background

Stochastic shortest path (SSP) problems extend the classic shortest path problems to stochastic settings. An SSP is defined by $M = \langle S, A, T, C, s_0, S_G \rangle$, where S is a finite set of states; A is a finite set of actions; $T(s, a, s') \in [0, 1]$ denotes the probability of reaching a state s' by executing an action a in state s; $C(s, a) \in \mathbb{R}^+ \cup \{0\}$ is the cost of executing action a in state s; $s_0 \in S$ is the initial state; and $S_G \subseteq S$ is the set of absorbing goal states. The cost of executing an action is positive in all states and it is zero in the goal states. SSPs generalize finite and infinite-horizon MDPs and have a discount factor $\gamma = 1$. The objective in an SSP is to minimize the expected cost of reaching a goal state from the start state. The optimal policy, π^*, can be extracted using the value function defined over the states, $V^*(s) = \min_a \quad Q^*(s, a), \ \forall s \in S$. The Q-value of the action a in state s is calculated as $Q^*(s, a) = C(s, a) + \sum_{s'} T(s, a, s')V^*(s')$.

A. Planning with Reduced Models

The complexity of solving SSPs optimally [13] has led to the use of approximation techniques such as reduced models. Reduced models simplify planning by considering a subset of outcomes, which is especially useful in problems with a high branching factor for action outcomes. Let $\theta(s, a)$ denote the set of all outcomes of (s, a), $\theta(s, a) = \{s' | T(s, a, s') > 0\}$.

Definition 1: A **reduced model** of an SSP M is represented by the tuple $M' = \langle S, A, T', C, s_0, S_G \rangle$ and characterized by an altered transition function T' such that $\forall (s, a) \in S \times A, \theta'(s, a) \subseteq \theta(s, a)$, where $\theta'(s, a) = \{s' | T'(s, a, s') > 0\}$ denotes the set of outcomes in the reduced model for action $a \in A$ in state $s \in S$.



(a) Problem Setting (b) ROS simulation

Fig. 2. An illustrative example of a robot navigating in a building, showing the problem setting (left) and its corresponding ROS simulation with black regions denoting obstacles and shaded region denoting human locations (right). S denotes start location and G is the goal location. Green line denotes the trajectory computed using determinization, red denotes optimal trajectory, blue denotes that of 0/1 RM.

We normalize the probabilities of the outcomes included in the reduced model. The outcome selection process in a reduced model framework determines the number of outcomes and how the specific outcomes are selected. Depending on these two aspects, a spectrum of reductions exist with varying levels of probabilistic complexity that ranges from the single outcome determinization to the full model.

An **outcome selection principle** (OSP) determines the outcomes included in the reduced model per state-action pair, and the altered transition function for each state-action pair. The OSP can be a simple function such as always choosing the most-likely outcome or a more complex function. Traditionally, a reduced model is characterized by a single OSP—a single principle is used to determine the number of outcomes and how the outcomes are selected across the entire model. Hence, existing reduced models are incapable of selectively adapting to risks. Fig. 2 illustrates this for a robot navigating in a building. Planning with determinization ignores the probability of encountering humans and therefore its optimal policy (green in 2(b)) conflicts with that of human trajectory, which can be avoided using the 0/1 RM approach (blue trajectory in 2(b)) described below.

III. $0/1$ Reduced Models

We present planning with a **p**ortfolio of **o**utcome **s**election **p**rinciples (POSP), a generalized approach to formulate risk-aware reduced models by switching between different OSPs. The approach is inspired by the benefits of using portfolios of algorithms to solve complex computational problems [14]. A *model selector* selects an outcome selection principle for each state-action pair.

Definition 2: Given a portfolio of finite outcome selection principles, $\mathrm{Z} = \{\rho_1, \rho_2, ..., \rho_k\}$, $k > 1$, a **model selector**, Φ, generates T' for a reduced model by mapping every (s, a) to an outcome selection principle, $\Phi : S \times A \rightarrow \rho_i$, $\rho_i \in \mathrm{Z}$, such that $T'(s, a, s') = T_{\Phi(s,a)}(s, a, s')$, where $T_{\Phi(s,a)}(s, a, s')$ denotes the transition probability corresponding to the outcome selection principle selected by the model selector.

In this work, we focus on a basic instantiation of POSP — $0/1$ RM — that switches between the extremes of outcome

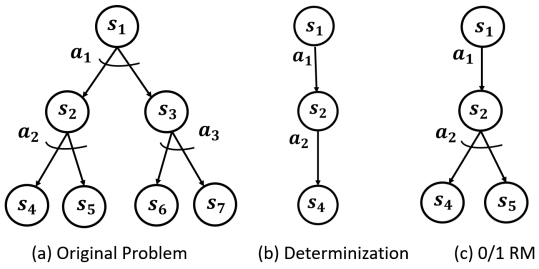


Fig. 3. Example reduced models formed with different techniques.

selection principles: determinization and the full model.

Definition 3: A **0/1 reduced model** (0/1 RM) is characterized by a model selector, $\Phi_{0/1}$, that selects either one or all outcomes of an (s, a) to be included in the reduced model.

In a 0/1 RM, the model selector that either ignores the stochasticity completely (0) by considering only one outcome of (s, a), or fully accounts for the stochasticity (1) by considering all outcomes of the state-action pair in the reduced model. An instantiation of 0/1 RM for the robot navigation in Fig. 2(a) uses the full model for the states around obstacles and regions with high probability of encountering humans.

Clearly, the existing reduced models, such as determinization, are a special case of POSP, with a model selector that always selects the same OSP for every state-action pair. In planning using a portfolio of OSPs, however, the model selector typically utilizes the synergy of multiple OSPs. Each state-action pair may have a different number of outcomes and a different mechanism to select the specific outcomes (Fig. 3). Hence, we leverage this flexibility in outcome selection to improve risk awareness in reduced models by using more informative outcomes in the risky states and using simple outcome selection principles otherwise. Though the model selector may use multiple OSPs to generate T' in a POSP, note that the resulting model is still an SSP. In this paper, we focus on creating reduced models that yield high quality results using the existing OSPs from the literature. Hence, future improvements in OSPs can be leveraged by POSPs. Depending on the model selector and the portfolio, a large spectrum of reduced models exists for an SSP and choosing the right one is non-trivial.

A. Model Selector (Φ)

The model selectors in existing reduced models have been devised typically to reduce planning time. An efficient Φ in a POSP optimizes the trade-off between solution quality and planning time. Devising an efficient model selector automatically can be treated as a meta-level decision problem that is computationally more complex than solving the reduced model, due to the numerous possible combinations of OSPs. Even in a 0/1 RM, devising an efficient Φ is non-trivial as it involves deciding when to use the full model and when to use determinization. We illustrate this with an example.

We experimented with a scaled-down version of the navigation example in Fig. 2(a), on a 5×5 grid, allowing only one full model usage, and using no oracle information. Therefore, the best 0/1 RM is identified by testing all valid formulations using most likely outcome determinization. Each state for this problem is represented as a tuple $\langle l, c\rangle$ where l denotes the location and c denotes if the robot is in conflict with a human or hits an obstacle. The robot can move in eight directions and the actions are stochastic, succeeding with a probability of 0.8, and cost $+10$ if $c = True$ and $+1$ otherwise. We generated models with different fidelities by altering when the full model is used. As expected, the highest gains are observed when the full model is used in states that lie on humans trajectories, with high probability. The time taken to identify the best setting with one full model usage is 2851 ms and involved solving for all possible reduced model formulation. Hence, for large real world problems with unconstrained full model use, exhaustive search in the space of models is not viable even for 0/1 RM.

In the worst case, all OSPs in Z may have to be evaluated to determine the best reduced model formulation for the more general setting. Let τ_{max} denote the maximum time taken for this evaluation across all states. When every action transitions to all states, the outcome selection principles in Z may be redundant in terms of the specific outcomes set produced by them. For example, selecting the most-likely outcome and greedily selecting based on heuristic could result in the same outcome for certain (s, a) pair. Using this, we show that the worst case complexity for a model selector is independent of the size of the portfolio, which may be very a large number in the worst case.

Proposition 1: The worst case time complexity for a model selector to generate T' for a POSP is $\mathcal{O}(|A|2^{|S|}\tau_{max})$.

Proof: For each (s, a), at most |Z| outcome selection principles are to be evaluated and this takes at most τ_{max} time (as mentioned above). Since this process is repeated for every (s, a), Φ takes $\mathcal{O}(|S||A||\mathrm{Z}|\tau_{max})$ to generate T'. In the worst case, every action may transition to all states and the outcome selection principles in Z may be redundant in terms of the specific outcomes set produced by them. Hence, the evaluation is restricted to the set of unique outcomes sets denoted by k, $|k| \leq |\mathcal{P}(S)|$, with $\mathcal{P}(S) = 2^{|S|}$. Then, it suffices to evaluate the $|k|$ outcome sets instead of |Z|, reducing the complexity to $\mathcal{O}(|A|2^{|S|}\tau_{max})$. ■

Corollary 1: The worst case time complexity for $\Phi_{0/1}$ to generate T' for a 0/1 RM is $\mathcal{O}(|A||S|^2\tau_{max})$.

Proof: This proof is along the same lines as that of Proposition 1. To formulate a $0/1$ RM of an SSP, it may be necessary to evaluate every outcome selection principle, $\rho_i \in$ Z, that corresponds to a determinization or a full model. Hence, in the worst case, $\Phi_{0/1}$ takes $\mathcal{O}(|S||A||\mathrm{Z}|\tau_{max})$ to generate T'. The set of unique outcomes, k, for a $0/1$ RM is composed of all unique deterministic outcomes, which is every state in the SSP, and the full model, $|k| \leq |S| + 1$. Replacing $|Z|$ with $|k|$, the complexity is reduced to $\mathcal{O}(|A||S|^2\tau_{max})$. ■

The current best approach to evaluate an OSP is to solve the corresponding reduced model and evaluate the policy in hindsight. Therefore, in the following section, we propose an approximation for outcome selection.

IV. Solution Approach

The first step in improving risk awareness of a reduced model is to identify the features that characterize risky states in the problem. In our running example (Fig. 2(a)), being in conflict with human trajectories or hitting an obstacle is a risk, which can be denoted by the state feature *c=true*. Since this cannot be estimated automatically by the agent without solving the full model, we formalize this as planning using information from an oracle (human). The agent queries an oracle which provides the features that characterize risks in a problem, denoted by $\vec{f_o}$. If problem instances in a domain are similar or share a structure in terms of state features, actions, and goal conditions, querying once per domain may suffice. For balancing the trade-off between model simplicity and risk awareness, it is more beneficial to use the full model in states that immediately lead to risky states primarily due to sub-optimal action selection that result from ignoring outcomes. We identify these states based on the values of the ignored outcomes for an action.

A. Reduction Impact

One of the reasons for reduced model techniques producing poor solutions is that some outcomes are completely ignored. The reduction impact δ is a measure of the values of ignored outcomes and is calculated for each (s, a). Following π^*, the reduction impact is calculated as, $\forall(s, a)$:

$$\delta(s,a) = Q^*(s,a) - \sum_{s' \in \theta'(s,a)} T'(s,a,s')V^*(s'). \qquad (1)$$

Therefore, the reduction impact is higher if risky states are ignored in the reduced model, due to their significantly higher expected cost of reaching a goal. Since the optimal values are unknown, we estimate this using samples. Sample trajectories generated by depth-limited random walk on the target problem or smaller problem instances from the domain may be used for this purpose. These samples are solved optimally and the reduction impacts corresponding to $\vec{f_o}$ are determined using these exact solutions. The reduction impacts for the given features are learned in hindsight by computing the mean values of the samples. More complex methods for aggregating the values from the samples may be considered. In our experiments, we generate samples by multiple trials of depth-limited random walk on the problem. The samples are solved using LAO* [15], which is an optimal solver based on A* [7] for solving MDPs with loops. We then learn the reduction impact with respect to a most likely outcome determinization of the problem. Clearly, as the number of samples and the depth of the random walk are increased, the estimates converge to their true values.

Optimal Reduction Impact For the class of problems described below, we show that δ can be calculated optimally, without using samples or having to solve the problem.

Consider an SSP in which an action can achieve a successful outcome with probability $1-p$ or fail with probability $p>0$. When an action fails, the state remains unchanged. Let s denote a state of the SSP for which a successful execution of action a with cost $C(s, a)$ results in outcome state s'. For problems with this structure, it has been shown that the Q-values can be calculated optimally as [16]:

$$Q^*(s,a) \;=\; \frac{C(s,a)}{1-p} + V^*(s').$$

Substituting the above equation in Equation 1, we get

$$\delta(s,a) = \frac{C(s,a)}{1-p}.$$

Thus, for problems with this structure, the reduction impact can be calculated optimally without using samples.

B. Outcome Selection Guided by Reduction Impact

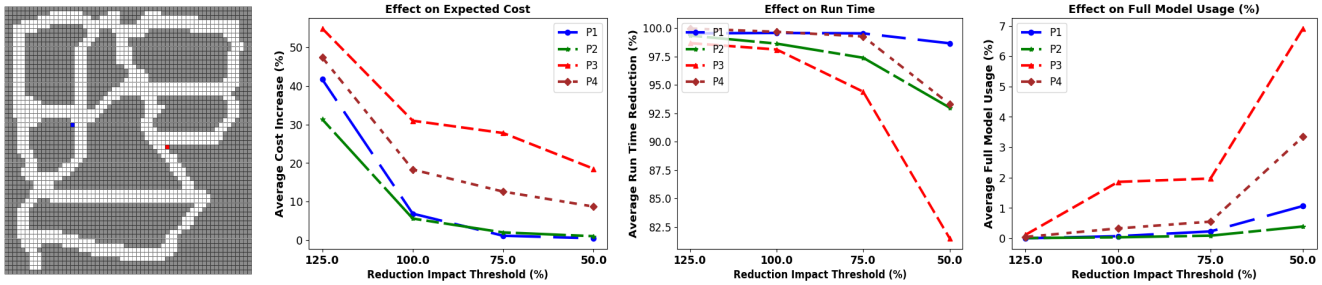
Since the reduction impact reflects the criticality of the states being ignored, we use this as a heuristic for model selection in a 0/1 RM. Ignoring risky outcomes in the reduced model results in an optimistic view of the problem, and hence a higher reduction impact. In a 0/1 RM, the full model may be employed in states with approximate reduction impact above a certain threshold and determinization in other states. By altering the δ threshold at which the full model is employed, reduced models with different levels of sensitivity to risks may be produced. This also produces reduced models with possibly different levels of computational gains and solution quality, due to the differences in fraction of full model usage. To demonstrate this, we compared the solution qualities of reduced models formed with different reduction impact thresholds (Fig. 4) for four instances of the racetrack domain [17][1]. The problems have the same actions, goal conditions, and state representation, but differ in the size of the state space. The threshold indicates the % difference between the estimated δ and the original costs, at which the full model is employed. In all other states, most likely outcome determinization is used. The cost increase is with respect to the lower bound (optimal expected cost obtained by solving the full model) and the run time reduction is with respect to solving the full model. The full model usage increases as the threshold lowers, resulting in improved solution quality (lower cost) and reduced run time savings.

V. Experimental Results

We evaluate 0/1 RM in three domains including an electric vehicle (EV) charging problem using real world data from a university campus, and two benchmark planning problems: the racetrack domain and the sailing domain. We compare 0/1 RM with the following reduced model techniques:

- Most-likely outcome determinization (MLOD);
- Uniformly selecting two outcomes greedily (M02)
- M_l^k reduction characterized by l primary outcomes and k exceptions, with $k=1, l=1$ [3]; and
- Reduced models, with $Z = \{\text{MLOD}, \text{M02}\}$, that alternate between MLOD and M02 (0/M02 RM).

[1]In racetrack domain, the objective is to move from start to goal states by applying acceleration and controlling the car correctly. If the robot (car) hits a wall, it is repositioned back to start state.



(a) Example racetrack problem (b) Effect on expected cost (c) Effect on run time (d) Effect on full model usage (%)

Fig. 4. Comparison of different reduction impact thresholds on four instances of the racetrack domain. In (a), blue denotes start state and red denotes goal, lighter cells denote the track and darker cells denote walls (obstacles).

The expected cost of reaching the goal and planning time are used as evaluation metrics. The features denoting risks were identified empirically. Given the features, the reduction impacts are estimated using thirty samples for each domain. For all states with reduction impact greater than the threshold value (Table I), the model selector uses a full model in a 0/1 RM and uses M02 reduction in 0/M02 RM. In all other states, MLOD is used. All results are averaged over 100 trials of planning and execution simulations and the average times include re-planning time. The deterministic problems are solved using the A* algorithm [7] and other problems using LAO*. All algorithms were implemented with $\epsilon{=}10^{-3}$ convergence and using h_{min} heuristic computed using a labeled version of LRTA* [18], and tested on an Intel Xeon 3.10 GHz computer with 16GB of RAM.

Racetrack Domain We experimented with six problem instances from the racetrack domain [17], in which the task is to move from start to goal states by applying acceleration correctly. If the car hits a wall, it is repositioned back to start state. We modified the problem to increase the difficulty such that, in addition to a 0.10 probability of slipping, there is a 0.20 probability of randomly changing the intended acceleration by one unit. The reduction impact uses one-step lookahead and state features such as: whether the successor is a wall or pothole or goal, and if the successor is moving away from the goal, estimated using the heuristic.

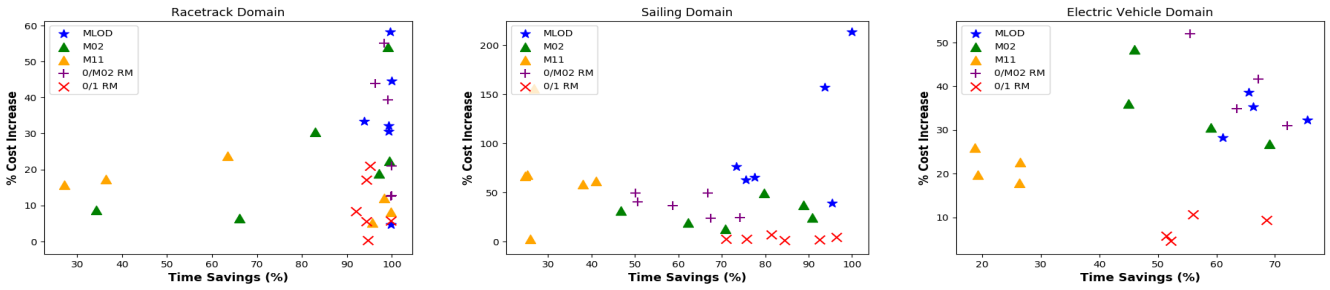
Sailing Domain We present results on six instances of the sailing domain [19], in which the objective is to find the shortest path between two points of a grid under fluctuating wind conditions. The boat cannot move in the direction opposite to that of the wind and the changes in wind direction are stochastic. The problem instances vary in terms of grid size and the goal position (opposite corner (C) or middle (M) of the grid). The reduction impact is estimated using one-step lookahead and based on state features such as: the difference between the action's intended direction of movement and the wind's direction, and if the successor is moving away from goal, estimated using the heuristic value.

EV Charging Problem We experimented with the electric vehicle (EV) charging domain, operating in a vehicle-to-grid setting, where the EV can charge and discharge energy from a smart grid [20]. The objective is to minimize the long-term operational cost of the EV, given the owner's charging preferences. We modified the original problem to allow for uncertainty regarding the parking duration of the EV, which is specified by a probability that certain states may become terminal states. The maximum parking duration is the horizon H. Each state is represented by $\langle l, t, d, p, e\rangle$, where l is the current charge level, $t \leq H$ is the time step, d and p denote the current demand and price distribution for electricity respectively, and $0 \leq e \leq 3$ denotes the anticipated departure time specified by the owner. If the owner has not provided this information, then $e = 3$ and the agent plans for H. Otherwise, e denotes the time steps remaining for departure. The process terminates when $t = H$ or if $e = 0$.

Each t is equivalent to 30 minutes in real time. We assume that the owner may depart between four to six hours of parking with a probability of 0.2 that they announce their planned departure time. Outside that window, there is a lower probability of 0.05 that they announce their departure. We experimented with four reward functions (RF). The rewards and the peak hours are based on real data [21]. The battery capacity and the charge speeds are based on Nissan Leaf configuration. We assume the charge and discharge speeds to be equal. The battery inefficiency is accounted for by adding a 15% penalty on the rewards. The reduction impact is estimated using state features: time remaining for departure, if the current time is peak hour, and if there is sufficient charge for discharging, with one step-lookahead.

EV Dataset The data used in our experiments consist of charging schedules of electric cars over a four month duration in 2017 from an American university campus. The data is clustered based on the entry and exit charges, and we selected 25 representative problem instances across clusters for our experiments. The data is from a typical charging station, where the EV is unplugged once the desired charge level is reached. Since we are considering an extended parking scenario as in a vehicle parked at work, we use parking duration (H) as 8 hours in our experiments.

Discussion Table I reports the full model usage (%) for 0/1 RM in our experiments, using reduction impact as heuristic for model selector. The best threshold values were identified empirically based on experiments such as those in Fig. 4. Fig. 5 shows the average increase in cost (%) and the time savings (%), with respect to solving the original problem optimally, of the five techniques. For the EV do-



(a) Racetrack (b) Sailing (c) EV

Fig. 5. Comparison of different reduced models on three domains. The time savings and cost increase are with respect to solving the problem (full model) optimally. Risk-aware reduced models have higher time savings and lower cost increase.

TABLE I

% FULL MODEL USAGE IN 0/1 RM USING REDUCTION IMPACT AS HEURISTIC FOR MODEL SELECTOR, CORRESPONDING TO δ THRESHOLD.

Problem	Full Model (%)	δ threshold (%)
EV-RF-1	7.644	120
EV-RF-2	9.956	120
EV-RF-3	8.989	120
EV-RF-4	9.852	120
Racetrack-P1	0.071	100
Racetrack-P2	0.034	100
Racetrack-P3	1.859	100
Racetrack-P4	0.327	100
Racetrack-P5	2.871	100
Racetrack-P6	0.860	100
Sailing-20(C)	37.414	120
Sailing-40(C)	37.478	120
Sailing-80(C)	37.495	120
Sailing-20(M)	37.414	120
Sailing-40(M)	37.478	120
Sailing-80(M)	37.495	120

main, the results are aggregated over 25 problem instances for each reward function. A low cost increase indicates that the performance of the technique is closer to optimal. A high time saving value indicates improved run time gains by using the model. Hence, the **lower right corner** of each image represents the most desired results.

We observe that 0/1 RM can effectively minimize the expected costs without significantly affecting run time and by sparingly using the full model. This indicates that using δ as a heuristic works well in practice. Furthermore, 0/1 RM performs consistently better than other techniques, in terms of the trade-off. We solve all the reduced models optimally using the same algorithm in order to assess the direct impact of model reduction. In practice, however, any SSP solver (optimal or not) may be used to further improve run time gains. Thus, the results demonstrate the benefits of our approach in formulating reduced models that balance the trade-off between model simplicity and risk awareness.

VI. CONCLUSION

We propose planning using a portfolio of outcome selection principles that provides flexibility in outcome selection for reduced models. We measure the reduction impact based on the ignored outcomes and describe how it can be used as a heuristic for model selector in POSP. Our empirical results demonstrate the promise of this framework, as this basic instantiation of a POSP improves performance without significantly affecting the run time gains. In the future, we aim to devise practical methods for automatically devising good model selectors beyond the reduction impact heuristic.

Acknowledgments: We thank Luis Pineda and Kyle Wray for fruitful discussions and feedback.

REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, "An analysis of stochastic shortest path problems," *Mathematics of Operations Research*, vol. 16, pp. 580–595, 1991.
[2] K. Gochev, B. Cohen, J. Butzke, A. Safonova, and M. Likhachev, "Path planning with adaptive dimensionality," in *SoCS*, 2011.
[3] L. Pineda and S. Zilberstein, "Planning under uncertainty using reduced models: Revisiting determinization," in *ICAPS*, 2014.
[4] B. Styler and R. Simmons, "Plan-time multi-model switching for motion planning," in *ICAPS*, 2017.
[5] S. Yoon, A. Fern, and R. Givan, "FF-Replan: A baseline for probabilistic planning," in *ICAPS*, 2007.
[6] T. Keller and P. Eyerich, "A polynomial all outcome determinization for probabilistic planning," in *ICAPS*, 2011.
[7] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, pp. 100–107, 1968.
[8] V. V. Unhelkar, C. Pérez-D'Arpino, L. Stirling, and J. A. Shah, "Human-robot co-navigation using anticipatory indicators of human walking motion," in *ICRA*, 2015.
[9] P. Santana, S. Thiébaux, and B. Williams, "RAO*: An algorithm for chance-constrained POMDPs," in *AAAI*, 2016.
[10] D. Kulić and E. A. Croft, "Safe planning for human-robot interaction," *Journal of Field Robotics*, vol. 22, pp. 383–396, 2005.
[11] M. Petrik and D. Subramanian, "An approximate solution method for large risk-averse markov decision processes," in *UAI*, 2012.
[12] S. Zilberstein, "Building strong semi-autonomous systems," in *AAAI*, 2015.
[13] M. L. Littman, "Probabilistic propositional planning: Representations and complexity," in *AAAI*, 1997.
[14] M. Petrik and S. Zilberstein, "Learning parallel portfolios of algorithms," *Annals of Mathematics and Artificial Intelligence*, vol. 48, pp. 85–106, 2006.
[15] E. A. Hansen and S. Zilberstein, "LAO*: A heuristic search algorithm that finds solutions with loops," *Artificial Intelligence*, vol. 129, pp. 35–62, 2001.
[16] E. Keyder and H. Geffner, "The HMDP planner for planning with probabilities," in *6th International Planning Competition*, 2008.
[17] A. G. Barto, S. J. Bradtke, and S. P. Singh, "Learning to act using real-time dynamic programming," *Artificial Intelligence*, vol. 72, pp. 81–138, 1995.
[18] R. E. Korf, "Real-time heuristic search," *Artificial Intelligence*, vol. 42, pp. 189–211, 1990.
[19] L. Kocsis and C. Szepesvári, "Bandit based Monte-Carlo planning," in *ECML*, 2006.
[20] S. Saisubramanian, S. Zilberstein, and P. Shenoy, "Optimizing electric vehicle charging through determinization," in *ICAPS Workshop on Scheduling and Planning Applications*, 2017.
[21] Eversource, "Eversource energy - time of use rates," https://www.eversource.com/clp/vpp/vpp.aspx, 2017.