EE736 Project - Paper Review

Karan Chadha

March 29, 2017

The topic I have chosen to explore for the assignment is Risk Constrained Markov Decision Processes and corresponding Reinforcement Learning ALgorithms with the risk metric being Conditional Value at Risk(CVaR). I have reviewed the following papers for this project:

1. Risk-Constrained Reinforcement Learning with Percentile Risk Criteria by Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson & Marco Pavone published in Journal of Machine Learning, 2016.
2. Risk-Constrained Markov Decision Processes by Vivek Borkar and Rahul Jain, published in IEEE Transaction on Automatic Control,2014

The structure of this assignment is as follows:

- Motivation for working on Risk-Constrained MDPs
- Description of the risk metrics: CVaR(Conditional Value at Risk) and VaR(Value at Risk)
- Description of the paper: Risk-Constrained Markov Decision Processes
- Description of the paper: Risk-Constrained Reinforcement Learning with Percentile Risk Criteria
- Conclusive remarks

Motivation

The most widely-adopted optimization criterion for Markov decision processes (MDPs) is represented by the risk-neutral expectation of a cumulative cost. However, in many applications one is interested in taking into account *risk*, i.e., increased awareness of events of small probability and high consequences. Generally speaking, *risk* can be taken into account in the following two ways:

- *Risk-Sensitive MDPs*: In this case, the risk is taken into account in the reward function itself and the objective is to maximize a risk-sensitive criterion such as the expected exponential utility, a variance related measure, the percentile performance conditional value-at-risk (CVaR). This in turn minimizes the risk.
- *Risk-Constrained MDPs*: In this case, the reward function is not altered and kept same as in a unconstrained MDP, but a cost function is introduced representing risk and it is constrained to some value. Examples of such cost functions are conditional value-at-risk(CVaR) of the cumulative cost, or as a chance constraint.

The papers I have reviewed for this assignment focus on problems related to Risk-Constrained MDPs.

Risk Metrics

Various risk metrics have been used in research related to Risk Constrained MDPs. We will focus on the VaR constraint(chance constraint) and the CVaR constraint. Let Z be a finite-mean $E[|Z|] < \infty$ random variable representing cost, with the cumulative distribution function $F_Z(z) = P(Z \leq z)$. The value-at-risk(VaR) at confidence level $\alpha \in (0, 1)$ is defined as

$$VaR_\alpha(z) = min\{z|F_Z(z) \geq \alpha\}$$

Here the minimum is attained because F_Z is non-decreasing and right-continuous in z. When F_Z is continuous and strictly increasing, $VaR_\alpha(z)$ is the unique z satisfying $F_Z(z) = \alpha$. VaR_α measures risk as the maximum cost that might be incurred with respect to a given confidence level α. This metric has many shortcomings including the fact that it is not subadditive.
The Conditional VaR (CVaR) was introduced by Artzner, et al., which is defined as, $E[Z|Z > \beta]$ where β is the VaR at level α. $CVaR_\alpha$ measures risk as the expected cost given that such cost is greater than or equal to VaR_α. Another equivalent definition is given by:

$$CVaR_\alpha(Z) \coloneqq \min_\nu \left\{\nu + \frac{1}{1-\alpha}\mathrm{E}[(Z-\nu)^+]\right\} \tag{1}$$

where $(x)^+ = \max(x, 0)$ represents the positive part of x.
Now, I will give some reasoning as to why the two definitions of $CVaR_\alpha$ are equivalent. For this I have referred to "Optimization of Conditional Value-at-Risk" by R. Tyrell Rockfeller and Stanislav Uryasev.

The Conditional expectation definition of $CVaR_\alpha$ can be written as

$$CVaR_\alpha = \frac{1}{1-\alpha}\int_{z>\beta} z\phi(dz)$$

Let

$$H_\alpha(Z,\nu) \coloneqq \left\{\nu + \frac{1}{1-\alpha}\mathrm{E}[(Z-\nu)^+]\right\}$$

Now,

$$\frac{\partial}{\partial\nu}H_\alpha(Z,\nu) = 1 + \frac{1}{1-\alpha}[F_Z(z) - 1]$$

$$= \frac{1}{1-\alpha}[F_Z(z) - \alpha]$$

Now, minima of $H_\alpha(Z,\nu)$ are attained when the partial derivatives w.r.t. ν is 0, i.e. $F_Z(z) = \alpha$. Let a value of ν where minima is achieved be ν_α. Now,

$$\begin{aligned} CVaR_\alpha = minH_\alpha(Z,\nu) = G_\alpha(Z,\nu_\alpha) &= \nu_\alpha + \frac{1}{1-\alpha}\int_{z>\nu_\alpha}(Z-\nu_\alpha)\phi(dz) \\ &= \nu_\alpha + \frac{1}{1-\alpha}\int_{z>\nu_\alpha} Z\phi(dz) - \nu_\alpha\frac{1}{1-\alpha}\int_{z>\nu_\alpha}\phi(dz) \\ &= \frac{1}{1-\alpha}\int_{z>\nu_\alpha} Z\phi(dz) \end{aligned}$$

Here, we have used $\int_{z>\nu_\alpha}\phi(dz) = 1-\alpha$, which is evident from the definition of ν_α. Hence, both the definitions of $CVaR_\alpha$ are equivalent. Moreover, it can also be shown that $CVaR_\alpha$ is a coherent risk measure i.e. it is monotonic, sub-additive, translation invariant and homogenous, wheras VaR_α isn't. Hence, using $CVaR_\alpha$ provides us some theoretical and computational advantages. Also, it is more useful to use $CVaR_\alpha$ when we are interested to study the cost in the tail of the risk distribution.

Risk-Constrained Markov Decision Processes

In this paper, the basic idea is to solve an MDP by maximizing the expected reward over a finite time horizon subject to an upper bound on ($CVaR_\alpha$) the conditional expectation of the total cost given the total cost exceeds some given level. Here, there is only one constraint on $CVaR_\alpha$ for the whole trajectory, rather than a different one for each time instant i.e. the issue of time consistency has been ignored.

Problem Formulation

Let us go over some notation. Consider $x \in X$ and $u \in U$ where x represents the state and u represents the control, a continuous reward function $r(x, u)$, a continuous cost function $c(x; u)$, a controlled transition function $P_u(dx', u)$ continuous in (x, u), and a finite horizon T. Time is discrete and starts at 0.We will denote a policy by $\mathbf{u} = u^T = (u_1, \ldots, u_T)$, where u_t is the control applied at time t according to this policy. It is assumed that only noisy observations of the cost are available. Thus, given a zero-mean i.i.d. noise process $\{\xi_t\}$ with strictly positive density ϕ a cumulative cost process $\{Y\}$ has been defined as

$$Y_0 = 0, Y_{t+1} = Y_t + c(X_t, u_t) + \xi_{t+1}$$

The conditional expectation definition of $CVaR_\alpha$ is considered and it is denoted by $\Xi_\alpha(Y)$. The problem is formulated using the bound on $CVaR_\alpha$ as C_α as follows:

$$\begin{array}{lll} \mathbf{rMDP}_\alpha & \min\limits_{u} & E[\sum_{t=0}^{T} r(X_t, u_t)] \\ & \text{s.t.} & \Xi_\alpha(Y_{T+1}) \leq C_\alpha \end{array} \tag{2}$$

Next, it is proved that for ξ_t $i.i.d$ with strictly positive density, a solution for $\mathbf{rMDP}_\alpha$ exists.

Solution and Algorithms

To make the problem tractable, it is assumed that the cost function is separable i.e. $c(x, u) = c(x) + c(u)$. The control at time T only affects future dynamics which are irrelevant since only finite horizon dynamics upto time T are considered. Thus, we set $u_T = \tilde{u}$ such that $c(\tilde{u}) = 0$. Then, for a given $VaR_\alpha = \beta_\alpha$ a state-value-at-risk function is defined as :

$$V_t(z) \coloneqq P(Y_{T+1} > \beta_\alpha | Z_t = z)$$

A backward recursive equation for the same with a fixed policy $\pi(z) = u$ can be written as

$$V_t(z) = \int \tilde{p}(dz'|z, u) V_{t+1}(z') \tag{3}$$

where $\tilde{p}(dz'|z, u)$ represents the transition kernel for the controlled Markov Chain $Z_t = (X_t, Y_t, v_t), t \geq 0$ with deterministic initial state and control. Here, $v_t = u_{t-1}$. Also, denote $\tilde{c}(z_t)$ as the cost accumulated upto time t. Then, it is proved that if $V_0(z_0) > 0$, then

$$\Xi_\alpha(Y) = \frac{1}{V_0(z_0)} E[\sum_{t=0}^{T} \tilde{c}(Z_t) V_t(Z_t)]$$

. Thus, the constraint on the MDP now is

$$\frac{1}{\alpha}E[\sum_{t=0}^{T}\tilde{c}(Z_t)V_t(Z_t)] \leq C_\alpha$$

.

An Offline Algorithm to solve rMDP$_\alpha$ (iRMDP)

To solve the **rMDP$_\alpha$** problem, the method of Lagrange Multipliers has been used and a dual variable λ has been introduced. The optimization problem is now given as:

$$\min_{\lambda \geq 0} \max_{u} E[\sum_{t=0}^{T} r(X_t, u_t) + \lambda(C_\alpha - \Xi_\alpha(Y_{T+1}))]$$

.

To solve the above equation a deterministic iterative algorithm is provided in which the following has been introduced:
$J_t(x, y)$: the optimal value function(expected sum of rewards).
$Q_t(z)$: introduced as an iterate for $CVaR_\alpha$ for a given β_α
The idea for the given algorithm is illustrated below:
What we wish to find is the optimal control sequence u_t. If the constraints are satisfied, it can be easily solved in a backward iteration by using $E[J_{t+1}]$ and $r(x, u)$ to maximise J_t. To be more precise, the equation is given below:

$$u_t = \arg\max_{u} \big(r(x, u) + E[J_{t+1}]\big)$$

Now, to ensure that the constraints are satisfied we need to simultaneously solve the problem of optimal λ. Also, note that the VaR_α value (β_α) is not known to us. To solve for these we use the iterative equations:

$$\beta^m_{n+1} = \beta^m_n - \gamma_n(\alpha - V_0(z_0))$$

$$\lambda_{m+1} = \beta_n - \eta_m(C_\alpha - Q^m_0(z_0))$$

Now, notice that, in the β update equation we require the value of $V_0(z_0)$ for current parameters. Since V_t is found by a backward recursive equation, we run the backward loop once to solve equation (3) for finding $V_0(z_0)$.
For the λ update, we need the value of $Q^m_0(z_0)$. For this, we need to solve the backward recursive equation for Q_t and it should be done for a converged value of β. Hence, for each value of λ, we want a converged value of β(hence it runs in an inner loop). Essentially we want β update to be on a slower timescale.

Intuition behind convergence: All the backward recursive equations are finite horizon and hence converge. The β update equation can be shown to track the following ODE:

$$\dot{\beta}(t) = \alpha - P(Y_{T+1} > \beta(t))$$

It can be shown that it converges to the stable equilbrium i.e. β^* such that $P(Y_{T+1} > \beta^*) = \alpha$. Next, the conergence of λ can be shown by standard sub-gradient techniques. If all the above values converge such that the constraints are satisfied, then the backward recursive equation in J_t converges to the optimal value

function and u_t update converges to optimal control sequence.

Now, this algorithm can suffer from the curse of dimensionality when T is large. In such cases function approximation(e.g. Linear) can be applied to V_t, Q_t, J_t to make the solution tractable.

An Online Algorithm to solve rMDP$_\alpha$ (oRMDP):
The online algorithm presented in the paper is very similar to the offline algorithm with the following distinguishing features:

- The state space and control space are now considered to be finite.
- The value function earlier is replaced by the Q-value function and the corresponding updates are of the Q-Learning type.
- All the updates are online and each expectation is replaced by an online update based on the states encountered and previous iteration value for that state.
- Instead of running iterates of β till convergence and then updating λ, a two timescale scheme has been adopted which can be proved to be equally effective.
- The updates for J_t are run on a slower timescale compared to even β and assuming all other variables to be quasi-static the convergence of the algorithm to optimal Q-Value function and hence optimal control sequence is proved.
- Another caveat for this algorithm is that all state-action pairs are to be sampled with non-zero relative frequency. This can be easily ensured by simply tweaking the action to be uniform random with probability ϵ and according to the update equations with probability $1 - \epsilon$

Comments:

The inclusions of all the updates in the innermost loops may not be necessary. The run-time can be reduced by bringing some of them to the outer loops. But, it really does not make much difference, since practically one will use the online version of the algorithm.

Risk-Constrained Reinforcement Learning with Percentile Risk Criteria

This paper has the following four contributions:

- Formulation of 2 Risk Constrained MDP problems with the $CVaR_\alpha$ constraint and chance (VaR_α) constraint and their solution using Lagrange multipliers. Also, for both problems, the Bellman optimality conditions w.r.t. an augmented MDP have been established.
- A trajectory-based policy gradient algorithm has been established for both $CVaR_\alpha$ constraint and chance constraint.
- Using the aforementioned Bellman optimality condition, actor-critic algorithms to optimize policy and value function approximation parameters in an online fashion have been derived.
- The effectiveness of the above algorithms has been portrayed in an optimal stopping problem as well as a personalized ad-recommendation problem.

The definition for $CVaR_\alpha$ from equation(1) has been used in this paper.

Problem Formulation

Let us go over some notation. A finite MDP is a tuple X, A, C, D, P, P_0 where $X = 1, \ldots, n$, x_{Tar} and $A = 1, \ldots, m$ are the state and action spaces, x_{Tar} is a recurrent target state, and for a state x and an action a, $C(x, a)$ is a cost function with $|C(x, a)| \leq C_{max}$, $D(x, a)$ is a constraint cost function with $|D(x, a)| \leq D_{max}$. A stationary policy $\mu(\cdot|x)$ for an MDP is a probability distribution over actions, conditioned on the current state. In policy gradient methods, such policies are parameterized by a k-dimensional vector θ, so the space of policies can be written as $\{\mu(\cdot|x;\theta), x \in X, \theta \in \Theta \subseteq R^k\}$. Let $T_{\mu,x}$ represent the first time we hit x_{Tar}. It is assumed that this time is uniformly bounded by some T. $\gamma in (0, 1]$ is defined as the discounting factor. Next, we define the following for a constant policy μ.

$\mathcal{C}^\theta(x) = \sum_{k=0}^{T-1} \gamma^k C(x_k, a_k)|x_0$ $\qquad$ $\mathcal{D}^\theta(x) = \sum_{k=0}^{T-1} \gamma^k D(x_k, a_k)|x_0$

$\mathcal{C}^\theta(x, a) = \sum_{k=0}^{T-1} \gamma^k C(x_k, a_k)|x_0, a_0$ $\qquad$ $\mathcal{D}^\theta(x, a) = \sum_{k=0}^{T-1} \gamma^k D(x_k, a_k)|x_0, a_0$

Next, the value function and the action value function are defined as follows:

$V^\theta(x) = E[\mathcal{C}^\theta(x)]$ $\qquad$ $Q^\theta(x, a) = E[\mathcal{C}^\theta(x, a)]$

Now, we define our optimization problem as follows: For $CVaR_\alpha$

$$\min_\theta V^\theta(x^0) \qquad subject \quad to \qquad CVaR_\alpha(\mathcal{D}^\theta(x^0)) \leq \beta$$

Equivalently,

$$\min_{\theta,\nu} V^\theta(x^0) \qquad subject \quad to \qquad H_\alpha(\mathcal{D}^\theta(x^0, \nu)) \leq \beta$$

For Chance constrained optimization:

$$\min_\theta V^\theta(x^0) \qquad subject \quad to \qquad P(\mathcal{D}^\theta(x^0) \geq \alpha) \leq \beta$$

Finally, two assumptions are stated which imply the existance of a smooth optimal policy which can be obtained using Lagrange Multipliers. Next, we apply the lagrangian relaxation procedure to reformulate the problems as follows:

$$\max_{\lambda \geq 0} \min_{\theta,\nu} \left(L(\nu, \theta, \lambda) := V^{\theta}(x^0) + \lambda(H_{\alpha}(\mathcal{D}^{\theta}(x^0, \nu)) - \beta) \right)$$

A similar reformulation can be done for the chance constraint.

A Trajectory-Based Policy Gradient Algorithm

CVaR Constraint:
The basic idea of the algorithm is to descend in (θ, ν) and ascend in λ using the gradients of $\mathrm{L}(\nu, \theta, \lambda)$. Salient features of the algorithm:

- Firstly, trajectories of the process are generated following the policy θ_i. These are used to estimate the gradients w.r.t. each parameters. These estimates can be proved to be unbiased estimates of the gradients using MCMC arguments.
- Each of the parameters are projected onto a smaller set which is compact and convex. This is necessary for the convergence of the algorithm.
- This is a multi-time scale algorithm where the updates of θ, ν, λ occur on different timescales. Again this is useful for proving convergence. Here, ν is updated on the fastest timescale, λ on the slowest.

A sketch of the proof of convergence:

- Convergence of ν update: ν being updated on the fastest timescale converges to a fixed point which assumes λ and θ to be constant(quasi-static). Convergence is proved by Lyapunov function arguments.
- Convergence of θ update: θ being updated on a faster timescale than λ but slower than ν converges to a fixed point which assumes λ to be constant(quasi-static) and θ to be equilibriated. Convergence is again proved by Lyapunov function arguments.
- Next step is to prove that θ^* and ν^* obtained from above two iterations are a local minimum for $\mathrm{L}(\theta, \nu, \lambda)$ for a given λ. Proof of this is pretty straightforward by contradiction.
- Convergence of λ update: λ being updated on the slowest timescale converges to a fixed point which assumes ν and θ to be equilibriated. Convergence is again proved by standard Lyapunov function arguments.
- Final step is to prove that θ^*, ν^* and λ^* obtained from above iterations form a local optimum of the function $\mathrm{L}((\theta, \nu, \lambda)$.

For the chance Constrained optimzation problem, we get two gradient update equations instead of three. The algorithm is very similar.

Actor-Critic Algorithms

CVaR Constraint:
The crux behind Actor-Critic algorithms in this formulation is to define a new augmented MDP the total discounted cost of which will take into account the constraint while finding the gradients. If this is possible, we can apply standard actor-critic methods to those variables whose gradients satisfy this.
Firstly, we define the augmented MDP as $\bar{M} = (\bar{X}, \bar{A}, \bar{C_\lambda}, \bar{P}, \bar{P_0})$. Here, $\bar{X} = X \times S$, action space remains same $\bar{P_0}(x, s) = P_0(x)I\{s_0 = s\}$ and

$$\bar{C_\lambda} = \begin{cases} \lambda(-s)^+/(1-\alpha) \text{ if } x = x_{Tar} \\ C(x,a) \text{ otherwise} \end{cases}$$

$$\bar{P}(x', s'|x, s, a) = \begin{cases} P(x'|x,a)(1)s' = (s - D(x,a))/\gamma \text{ if } x \in X' \\ (1)x' = x_{Tar}, s' = 0 \text{ if } x = x_{Tar} \end{cases}$$

Now, using this, we can define $V^\theta(x^0, \nu)$ as

$$V^\theta(x^0, \nu) = E[\mathcal{C}^\theta(x^0) + \frac{\lambda}{1-\alpha}(\mathcal{D}^\theta(x^0) - \nu)^+]$$

Now, we can see that $\nabla_\theta L(\theta, \nu, \lambda) = \nabla_\theta V^\theta(x^0, \nu)$ Now, using Linear function approximation for the value function, standard actor critic methods using Temporal Differences can be applied to find the θ update equations.
Similarly, the update equations for λ and ν can be written in terms of this augmented MDP by relating the gradient of $L(\theta, \nu, \lambda)$ with $V^\theta(x^0, \nu)$. Now, in these algorithms, we want the updates to be online instead of trajectory based. For this we follow our policy and update parameters in accordance with the samples observed. An issue here, in fact in all actor-critic based schemes is that the estimate for gradient is unbiased only if states are sampled according to the discounted occupation measure of the states, which is not the case here.
Other than this, for the ν update, two alternatives have been provided: One is to update it online like all other parameters using SPSA to estimate the gradient using current value function of MDP estimates. Another alternative is to update ν only at the end of each trajectory where we won't be making use of the augmented MDP. The second method is preferred sometimes due to high numerical values of the Gradient estimates using SPSA as Δ goes to 0.
Convergence proofs in this case involve the use of the Contraction property of the Bellman Operator, which we can define now for the augmented MDP. Some parts of the proof like convergence of all to a local optima are similar to the policy gradient method proofs.

The actor critic method for the chance constrained MDP is very similar to that of CVaR constrained. We first define an augmented MDP with a cost function:

$$\bar{C_\lambda}(x, s, a) = \begin{cases} \lambda(1)\{s \leq 0\} \text{ if } x = x_{Tar} \\ C(x,a) \text{ otherwise} \end{cases}$$

The rest of the analysis is very similar to the CVaR constrained problem.

Examples

Optimal Stopping problem: The state at each time step k $\leq$ T consists of the cost c_k and time k, i.e., x = (c_k, k), where T is the stopping time. The agent (buyer) should decide either to accept the present cost ($u_k = 1$) or wait ($u_k = 0$). If he/she accepts or when k = T, the system reaches a terminal state and the cost max(K, c_k) is received (K is the maximum cost threshold), otherwise, she receives a holding cost p_h and the new state is (c_{k+1}, k + 1). This problem can be reformulated as:

$$\min_{\theta} E[\mathcal{C}^{\theta}(x^0)] \qquad subject\ \ to \qquad CVaR_{\alpha}(\mathcal{C}^{\theta}(x^0)) \leq \beta\ or\ P(\mathcal{D}^{\theta}(x^0) \geq \alpha) \leq \beta$$

This can now be solved using the methods developed in this paper. On running experiments on this problem the various observations suggested that the risk-constrained algorithms yield a higher expected cost, but less worst-case variability, compared to the risk-neutral methods. More precisely, the cost distributions of the risk-constrained algorithms have lower right-tail (worstcase) distribution than their risk-neutral counterparts. It was noticed that while the risk averse policy satisfies the CVaR constraint, it is not tight (i.e., the constraint is not matched). However, since both the expectation and CVaR risk metrics are subadditive and convex, one can always construct a policy that is a linear combination of the risk neutral optimal policy and the risk averse policy, such that it matches the constraint threshold and has a lower cost compared to the risk averse policy.

Personalized Ad-Recommendation Systems: The goal here is to generate a strategy that for each user of the website selects an ad that when it is presented to her has the highest probability to be clicked on. The methods currently empoyed in modeling this system does not distinguish between a visit and a visitor. If we distinguish, we need a new class of solutions as provided by this paper. The problem can be formulated as:

$$\min_{\theta} E[\mathcal{R}^{\theta}(x^0)] \qquad subject\ \ to \qquad CVaR_{1-\alpha}(-\mathcal{R}^{\theta}(x^0)) \leq \beta$$

Now, the methods developed in this paper can be applied. The results indicate that the risk-constrained algorithms yield a lower expected reward, but have higher left tail (worst-case) reward distributions.

Conclusive Remarks

To give an overall brief summary of the above two papers, one can say that a variant of constrained MDPs i.e. Risk-Constrained MDPs has been formulated. The analogs of basic algorithms to solve any MDP are discussed in the first paper whereas how these methods can be used in the real world under Reinforcement Learning techniques is discussed in the second paper.