Rounding a ROBP Using DP

Prithviraj Ghosh

May 2025

1 Proposal

Quoting from the ROBP paper "We can construct M^0 from M in time $O\left(n^2 \log(W) \log\left(\frac{n}{\varepsilon}\right)/\varepsilon\right)$."
I propose using the dp algorithm learnt from the second paper we can construct M^0 from M in time $O\left(n^3 \log\left(\frac{n}{\varepsilon}\right)/\varepsilon\right)$

2 Proof

2.1 Idea

According to the claim 3.2 of the ROBP paper "Each vertex $v_j \in L(M^i, i+1)$ can be computed in time $O\left(\log\left(\frac{n}{\varepsilon}\right)\log W\right)$." I try to reduce it to $O\left(\log\left(\frac{n}{\varepsilon}\right)\right)$ using a precomputed dp table as used in the other paper. We will use a similar definition of τ and T appropriately for doing the approximation.

2.2 Defining τ

Consider we have created M^{i+1} and creating M^i and like the assumption of the claim "We have the vertices v_j of $L(M^i, i+1)$ stored in a binary tree, and also know their acceptance probabilities $P_{M^i}(\cdot)$." So by the method of rounding used to create M^i 0 must be the first vertex in layer i and the probability of which can be calculated in $O\left(\log\left(\frac{n}{\varepsilon}\right)\right)$ as discussed in the paper. Now for the upcoming vertices in the layer we will use the following definition of τ :

$$\tau : \{0, \ldots, n\} \times \mathbb{R}_{\geq 0} \to \mathbb{R} \cup \{\pm\infty\}$$

where $\tau(i, a)$ is defined as is the smallest C such that there exist at least a solutions to the knapsack problem with weights $w_{i+1}, \ldots, w_n$ and capacity C which is the same definition as the paper just with different weights as the orientation of the weights did not matter in the analysis later

Claim: Assume we have calculated v_j in the layer i along with its probability to reach an accepting state p and want to find v_{j+1} then the following equation holds exactly

$$v_{j+1} = C - \tau\left(i, \frac{2^{n-i}p}{1+\varepsilon}\right) + 1$$

Proof: By the definition of τ the expression $C - \tau\left(i, \frac{2^{n-i}p}{1+\varepsilon}\right)$ is the maximum v such that the probability of v to reach the accepting state is at least $\frac{p}{1+\varepsilon}$. So by adding 1 we get exactly v_{j+1} by the algorithm of rounding discussed in the paper.

2.3 Approximating τ

Consider we have the T table as mentioned in the paper and creating that took time $O\left(n^3 \log\left(\frac{n}{\varepsilon}\right)/\varepsilon\right)$ then we can take the value $T\left[i, \left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor\right]$ as an appropriate approximation of $\tau\left(i, \frac{2^{n-i}p}{1+\varepsilon}\right)$ as by the lemma 2.2 of the dp paper

$$T\left[i, \left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor\right] \leq \tau\left(i, Q^{\left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor}\right) \leq \tau\left(i, \frac{2^{n-\ell}p}{1+\varepsilon}\right)$$

and

$$\begin{aligned} T\left[i, \left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor\right] &\geq \tau\left(i, Q^{\left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor - n + i}\right) \\ &\geq \tau\left(i, Q^{\log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right) - n + i - 1}\right) \\ &= \tau\left(i, \frac{2^{n-\ell}p}{Q^{n-i+1}(1+\varepsilon)}\right) \\ &\geq \tau\left(i, \frac{2^{n-\ell}p}{(1+\varepsilon)^2}\right) \approx \tau\left(i, \frac{2^{n-\ell}p}{1+2\varepsilon}\right) \end{aligned}$$

So

$$\tau\left(i, \frac{2^{n-\ell}p}{1+2\varepsilon}\right) \leq T\left[i, \left\lfloor \log_Q\left(\frac{2^{n-i}p}{1+\varepsilon}\right)\right\rfloor\right] \leq \tau\left(i, \frac{2^{n-\ell}p}{1+\varepsilon}\right)$$

Hence by putting it in the claim in above we get the approximate v_{j+1} calling it $(v_{j+1})_{\text{approx}}$ lies between the two approximations always

$$(v_{j+1})_{\varepsilon} \leq (v_{j+1})_{\text{approx}} \leq (v_{j+1})_{2\varepsilon}$$

Therefore, by monotonicity of the probability

$$\frac{p}{1+\varepsilon} \geq p_{v_{j+1}} \geq \frac{p}{1+2\varepsilon}$$

Calling $p_{v_{j+1}} = \frac{p}{1+k\varepsilon}$ we get a $k \in [1,2]$, storing this k along with the probability value of v_{j+1} we can now complete M^0 with the approximation of $k\varepsilon$ and since k is a constant in [1,2] it is similar to the given approximation in the paper with time complexity $O\left(n^3 \log\left(\frac{n}{\varepsilon}\right)/\varepsilon + n^2 \log\left(\frac{n}{\varepsilon}\right)/\varepsilon\right) = O\left(n^3 \log\left(\frac{n}{\varepsilon}\right)/\varepsilon\right)$