IE 617: Online Learning and Bandit Algorithms Course Project
Communication-Constrained Multi-Armed Bandits

Adway Girish, Fathima Zarin Faizal
(180070002, 180070018)

Department of Electrical Engineering
IIT Bombay

October 30, 2022

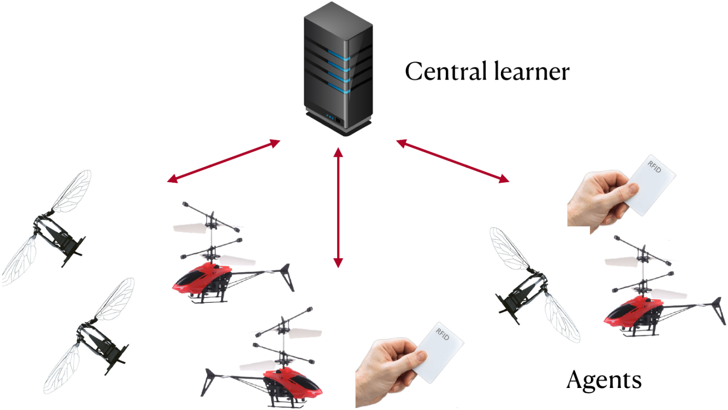
Outline

1. Pre-Project Recap
2. Theorems and Proofs
3. Simulations
4. Conclusion

Applications of Learning to Communication

- Beam alignment (Vutha Va, Takayuki Shimizu, Gaurav Bansal, et al. "Online Learning for Position-Aided Millimeter Wave Beam Training". In: *IEEE Access* (2019), Matthew B. Booth, Vinayak Suresh, Nicolò Michelusi, et al. "Multi-Armed Bandit Beam Alignment and Tracking for Mobile Millimeter Wave Communications". In: *IEEE Communications Letters* 7 (2019))
- Rate selection (Harsh Gupta, Atilla Eryilmaz, and R. Srikant. "Link Rate Selection using Constrained Thompson Sampling". In: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 2019)
- Bit-constrained communication (Osama A. Hanna, Lin F. Yang, and Christina Fragouli. *Solving Multi-Arm Bandit Using a Few Bits of Communication*. 2021, Aritra Mitra, Hamed Hassani, and George J Pappas. "Linear Stochastic Bandits over a Bit-Constrained Channel". In: *arXiv preprint arXiv:2203.01198* (2022))

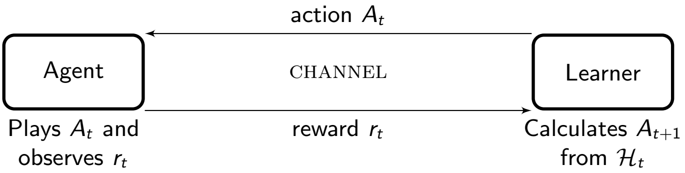
Problem Setup



Source: Osama A. Hanna, Lin F. Yang, and Christina Fragouli. *Solving Multi-Arm Bandit Using a Few Bits of Communication*. 2021

Problem Statement

- MAB problem, horizon n
- Leaner chooses $A_t \in \mathcal{A}_t$ and receives r_t with mean μ_{A_t}
- Goal: maximize expected regret, $R_n = \mathbb{E}[\sum_{t=1}^{n}(\mu_t^* - r_t)]$, where $\mu_t^* = \max_{A \in \mathcal{A}_t} \mu_A$



Recall

Let n be the number of rounds.

- ETC and ϵ-greedy achieves $\mathcal{O}(\sqrt{n})$ with knowledge of Δ
- Thompson sampling and UCB achieves $\mathcal{O}(\sqrt{n \log n})$ without knowing Δ
- LinUCB achieves $\mathcal{O}(d\sqrt{n} \log n)$

These assumed full-precision rewards.

Goal: Develop quantization scheme to apply over *any* MAB algorithm such that the quantized regret is only a constant factor off, while maintaining a low number of bits

Quantization

$\mathcal{L}$: countable set

Quantizer consists of:

- $\mathcal{E} : \mathbb{R} \to \mathcal{L}$
- $\mathcal{D} : \mathcal{L} \to \mathbb{R}$

Stochastic Quantization

Let $\mathcal{L} = \{\ell_i\}_{i=1}^{2^B}$, $x \in [\ell_1, \ell_{2^B}]$.

- $i(x) = \max\left\{j \mid \ell_j \leq x \text{ and } j < 2^B\right\}$
- $\mathcal{E}_{\mathcal{L}}(x) = \begin{cases} i(x) & \text{with probability } \frac{\ell_{i(x)+1} - x}{\ell_{i(x)+1} - \ell_{i(x)}} \\ i(x)+1 & \text{with probability } \frac{x - \ell_{i(x)}}{\ell_{i(x)+1} - \ell_{i(x)}} \end{cases}$
- $D_{\mathcal{L}}(j) = \ell_j, j \in \left\{1, \ldots, 2^B\right\}$

Conditioned on A_t, unbiased estimate of μ_{A_t} is communicated.

QuBan

- Maintains Markov property, unbiasedness, bounded variance for quantized rewards
- Uses a few bits for communication

QuBan: Main Ideas

- Center quantization scheme around value believed to be closest to picked arm's mean in majority of iterations
- Quantization error conditionally independent on past history given A_t
- Assign shorter codes to values near quantization centre and o.w. longer codes
- Use SQ to convey unbiased estimate of reward

QuBan: Algorithm (Learner)

Algorithm 1 Learner operation with input MAB algorithm Λ

```
Initialize: μ̂(1) = 0
for t = 1, ..., n do
    Choose an action A_t based on the bandit
     algorithm Λ and ask the next agent to play it
    Send M_t[6], μ̂(t) to an agent
    Receive the encoded reward (b_t, I_t, E_{L_t}(e_t)) (see
     Algorithm 2)
    Decode r̂_t:
    if length(b_t) ≤ 4 then
        r̂_t can be decoded using a lookup table
    else
        Decode the sign, s_t, of r_t from b_t
        Set ℓ_t to be the I_t-th element in the set
        {0, 2^0, ...}
        Set L_t = {ℓ_t, ℓ_t + 1, ..., max{2ℓ_t, ℓ_t + 1}}
        Let e_t^(q) = D_{L_t}(E_{L_t}(e_t))
        r̂_t = (s_t(e_t^(q) + ℓ_t + 3.5) + 0.5 + ⌊μ̂(t)/M_t⌋)M_t
    Calculate μ̂(t + 1) (using one of the discussed
     choices)
    Update the parameters required by Λ
```

Source: Osama A. Hanna, Lin F. Yang, and Christina Fragouli. *Solving Multi-Arm Bandit Using a Few Bits of Communication*. 2021

QuBan: Algorithm (Agent)

Algorithm 2 Distributed Agent Operation

Inputs: r_t, $\hat{\mu}(t)$ and M_t

Set $L = \{\lfloor \bar{r}_t \rfloor, \lceil \bar{r} \rceil\}$, $\hat{\bar{r}}_t = D_L(\mathcal{E}_L(\bar{r}_t))$

Set b_t with three bits to distinguish between the 8 cases: $\hat{\bar{r}}_t < -2, \hat{\bar{r}}_t > 3, \hat{\bar{r}}_t = i, i \in \{-1, 0, 1, 2\}$.

if $|\hat{\bar{r}}_t| > |a|$ and $\hat{\bar{r}}_t a > 0$, $a \in \{-2, 3\}$ **then**

  Augment b_t with an extra one bit to indicate if $|\hat{\bar{r}}_t| = |a| + 1$ or $|\hat{\bar{r}}_t| > |a| + 1$.

  if $|\hat{\bar{r}}_t| > |a| + 1$ **then**

    Let $L' = \{0, 2^0, ...\}$

    Set $\ell_t = \max\{j \in L | j \leq |\bar{r}_t| - |a|\}$

    Encode ℓ_t by $I_t - 1$ zeros followed by a one

    (unary coding), where I_t is the index of ℓ_t

    in the set L'.

    Let $e_t = |\bar{r}_t| - |a| - \ell_t$

    Set $\mathcal{L}_t = \{\ell_t, \ell_t + 1, ..., \max\{2\ell_t, \ell_t + 1\}\}$

    Encode e_t using SQ to get $\mathcal{E}_{\mathcal{L}_t}(e_t)$

Transmit $(b_t, I_t, \mathcal{E}_{\mathcal{L}_t}(e_t))$

Source: Osama A. Hanna, Lin F. Yang, and Christina Fragouli. *Solving Multi-Arm Bandit Using a Few Bits of Communication*. 2021

Assumptions on MAB Instance and Algorithm

Assumption 1

All codes are prefix-free codes. Further,

1. *rewards possess Markov property; and*
2. *the expected regret is upper-bounded by R_n^U.*

Regret Bound

Proposition 1

Suppose Assumption 1 holds. Then, when we apply QuBan, *the following hold:*

1. *Conditioned on* A_t, *the quantized reward* $\hat{r}_t$ *is* $\left(\left(1+\frac{\epsilon}{2}\right)\sigma\right)^2$*-subgaussian, conditionally independent on the history* $A_1, \hat{r}_1, \ldots, A_{t-1}, \hat{r}_{t-1}$ *(Markov property), and satisfies* $\mathbb{E}\left[\hat{r}_t \mid A_t\right] = \mu_{A_t}, |\hat{r}_t - r_t| \leq M_t$ *almost surely* $(t = 1, \ldots, n)$.
2. *The expected regret* R_n *is bounded as* $R_n \leq \left(1+\frac{\epsilon}{2}\right) R_n^U$, *where* ϵ *is a parameter to control the regret vs number of bits trade-off.*

Number of Bits

Theorem 1

Suppose Assumption 1 holds. Let $\epsilon = 1$. There is a universal constant C such that, for QUBAN *with:*

1. $\hat{\mu}(t) = \hat{\mu}_{A_t}(t-1)$ *(avg-arm-pt), the average number of bits communicated satisfies that* $\mathbb{E}[\bar{B}(n)] \leq 3.4 + \frac{C}{n}\sum_{i=1}^{k} \log\left(1 + |\mu_i|/\sigma\right) + C/\sqrt{n}$.
2. $\hat{\mu}(t) = \frac{1}{t-1}\sum_{j=1}^{t-1} \hat{r}_j$ *(avg-pt), the average number of bits communicated satisfies* $\mathbb{E}[\bar{B}(n)] \leq 3.4 + \frac{C}{n}\left(1 + \log\left(1 + \frac{|\mu^*|}{\sigma}\right) + \frac{R_n}{\sigma} + \sum_{t=1}^{n-1} \frac{R_t}{(\sigma t)}\right) + C/\sqrt{n}$.

Lower Bound

Theorem 2

For any memoryless algorithm that only uses quantized rewards, prefix-free encoding and satisfies that for any MAB instance with subgaussian rewards:

1. R_n *is sublinear in* n,
2. *Conditioned on* r_t, $\hat{r}_t - r_t$ *is* $\left(\frac{\sigma}{2}\right)^2$*-subgaussian* $(t = 1, \ldots, n)$,

there exist σ^2*-subgaussian reward distributions for which:*

1. $(\forall b \in \mathbb{N})(\exists t, \delta > 0)$ *such that* $\mathbb{P}\left[B_t > b\right] > \delta$.
2. $(\forall t > 0)(\exists n > t)$ *such that* $\mathbb{E}[\bar{B}(n)] \geq 2.2$ *bits.*

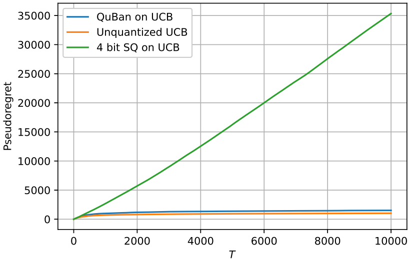
Upper Bound

$$
\begin{aligned}
B_t \leq 3 &+ \mathbf{1}\left[\frac{r_t}{M_t} - \left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor > 3\right] + \mathbf{1}\left[\left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor - \frac{r_t}{M_t} > 2\right] \\
&+2\left(\mathbf{1}\left[\frac{r_t}{M_t} - \left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor > 4\right] \mid \log\left(\frac{r_t}{M_t} - \left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor - 3\right)\right]\right) \\
&+2\left(\mathbf{1}\left[\left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor - \frac{r_t}{M_t} > 3\right]\left\lceil \log\left(\left\lfloor \frac{\hat{\mu}(t)}{M_t} \right\rfloor - \frac{r_t}{M_t} - 2\right)\right]\right) B_t \leq \quad 3 + \\
&+2\left(\mathbf{1}\left[\left|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}\right| > 3\right] \log\left(\left|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}\right| - 2\right)\right) B_t \leq \quad 3 + \\
&+2\left(\mathbf{1}\left[\left|\frac{r_t - \mu_{A_t}}{\sigma}\right| > 3(1-\delta)\right] + \mathbf{1}\left[\left|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}\right| > 3\delta\right]\right) \\
&+2\left(\mathbf{1}\left[\left|\frac{r_t - \mu_{A_t}}{\sigma}\right| > 3\right]\right) \log\left(\left|\frac{r_t - \hat{\mu}(t)}{\sigma}\right| - 2\right) \quad \text{for each } \delta > 0
\end{aligned}
$$

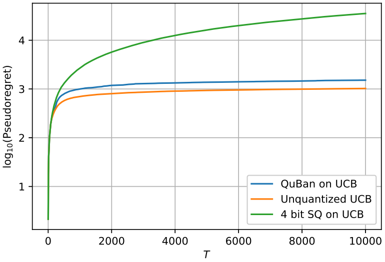
Upper Bound

$$
\begin{aligned}
\mathbb{E}\left[B_t\right] \leq 3 &+ \mathbb{P}\left[\left|\frac{r_t - \mu_{A_t}}{\sigma}\right| > 2(1-\delta)\right] + \mathbb{P}\left[\left|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}\right| > 2\delta\right] \\
&+ 2\left(\mathbb{P}\left[\left|\frac{r_t - \mu_{A_t}}{\sigma}\right| > 3(1-\delta)\right] + \mathbb{P}\left[\left|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}\right| > 3\delta\right]\right) \\
&+ 2\mathbb{E}\left[\left(\mathbf{1}\left[\left|\frac{r_t - \mu_{A_t}}{\sigma}\right| > 3\right]\right)\log\left(\left|\frac{r_t - \hat{\mu}(t)}{\sigma}\right| - 2\right)\right] \\
\leq 3.4 &+ C\mathbb{E}\left[\left|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}\right|\right] \leq \cdots \qquad \square
\end{aligned}
$$

QuBan



(a) Pseudoregret vs T



(b) $\log_{10}(\text{Pseudoregret})$ vs T

Pseudoregret vs. T for QuBan

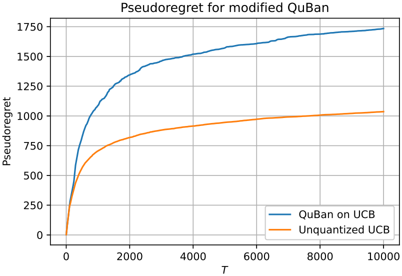
Modified setup

- Agent full precision, learner bit-constrained? Trivial.
- Both bit-constrained?

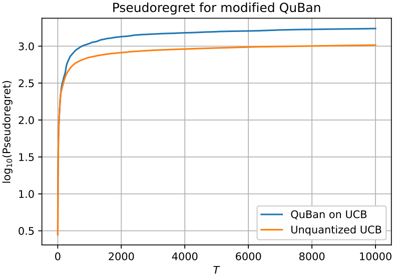
Modified QuBan

- Learner too is communication-constrained
- Learner sends $\hat{\mu}(t)$ using 10-bit SQ

Modified QuBan



(a) Pseudoregret vs T



(b) $\log_{10}(\text{Pseudoregret})$ vs T

Pseudoregret vs. T for modified QuBan

Conclusion

- Presented the upper bound proof, and
- Numerical analysis for the setup where both the learner and agents are bit-constrained.

Thank you