BTP- Summary

A1

Algo:

for t in 1 to T :

- → Select encoder from lambda $(1 \times en)$
- → Adversary selects channel
- → get b from E (error matrix) $(en \times ch)$
- → calculate tprob[c] - prob. of selecting channel c from b
- → Now update temp[i,j] - matrix of temporary transition probabilities using eqn.

  Δtemp[i,j] = lo (sprob[i] tprob[j])

  sprob ↑ steady state prob. from target transition matrix (P)

- → if t % step = 0 :
  - → update P - target trans. prob. with temp
  - → calculate sprob
  - → Do linprog using this new sprop, E, rate and update lambda

Notes:

→ Two transition matrices are used P(target), temp(temporary) where target is updated for every few steps to reduce variance

A2

Algo:

for t in 1 to T :

→ Select encoder from lambda
(1xen)

→ Adversary selects channel

→ get b from E (error matrix)
(enxch)

→ if t % step = 0 :

→ model a multinomial HMM with

transition matrix - initialized to P
(last estimate)

start prob. - steadyprob(P)

emission prob : lamda * E (enxch)
(1xch) (1xen)
(last estimate)

feed the HMM with b and train transition matrix

→ update P with lr and trans_mat

$\Delta P = lr(\text{trans mat} - P)$

→ calculate sprob

→ Do linprog using this new sprop, E, rate and update lambda

Notes:

Notes:

→ Do eps-Greedy selection of encoders and decrease learning rate over time to get better estimates on p

A3 :

Non Markov OPLB - Approx:

Algo: Same as the one in the paper with few sidelines:

→ Instead of solving quadratic constrain used x_{t-1} in the square term

→ used $\alpha_c = 0.001$ but given $\alpha_c \geq 1$ as $\alpha_c \geq 1$ is making equations infeasible for some initial values of x (= $lambda^T$)

→ Also the constrain part can be loosned as our θ^* is known

→ used pseudo inverse in some cases instead of inverse as it is blowing up for almost singular matrices which is often the case

Non Markov OPMD - Approx:

Algo: from the same paper

Sidelines:

→ Consider each encoder as an arm now π (= $lamda^T$)

→ Even here used $\alpha_c = 0.001$ otherwis linprog. equations are becoming infesable

NonMarkov OPLB - No approx - using cvxopt modelling

Algo: As it is mentioned in the paper with no approximation

Notes: → Rate is converging to lower value than optimal / any of the above algorithms.

Markov OPLB - Approx

Algo:

for t from 1 to T:

→ Select encoder from lambda (1xen)

→ Adversary selects channel

→ get b from E (error matrix) (enxch)

→ if t % step = 0 :

→ model a multinomial HMM with

transition matrix - initialized to P (last estimate)

start prob. - steady prob(P)

emission prob (1xch) : lamda (1xen) (last estimate) * E (enxch)

feed the HMM with b and train transition matrix

→ update P with lr and trans_mat

$\Delta P = lr\,(\text{trans mat} - P)$

→ Now calulate cost as

some huristic of

$$y = P_{(ch \times ch)} * E'_{(ch \times en)} * x_{(en \times 1)}$$

* here $y[i]$ is the cost if previous channel is i
* using max (y) will get safe x

→ Now apply OPLB as usual using this cost formulation.