ADAPTATIVE INTERIOR-POINT METHODS

APPLIED CONVEX OPTIMIZATION (ACO)

TSC - UPC

SPRING 2023

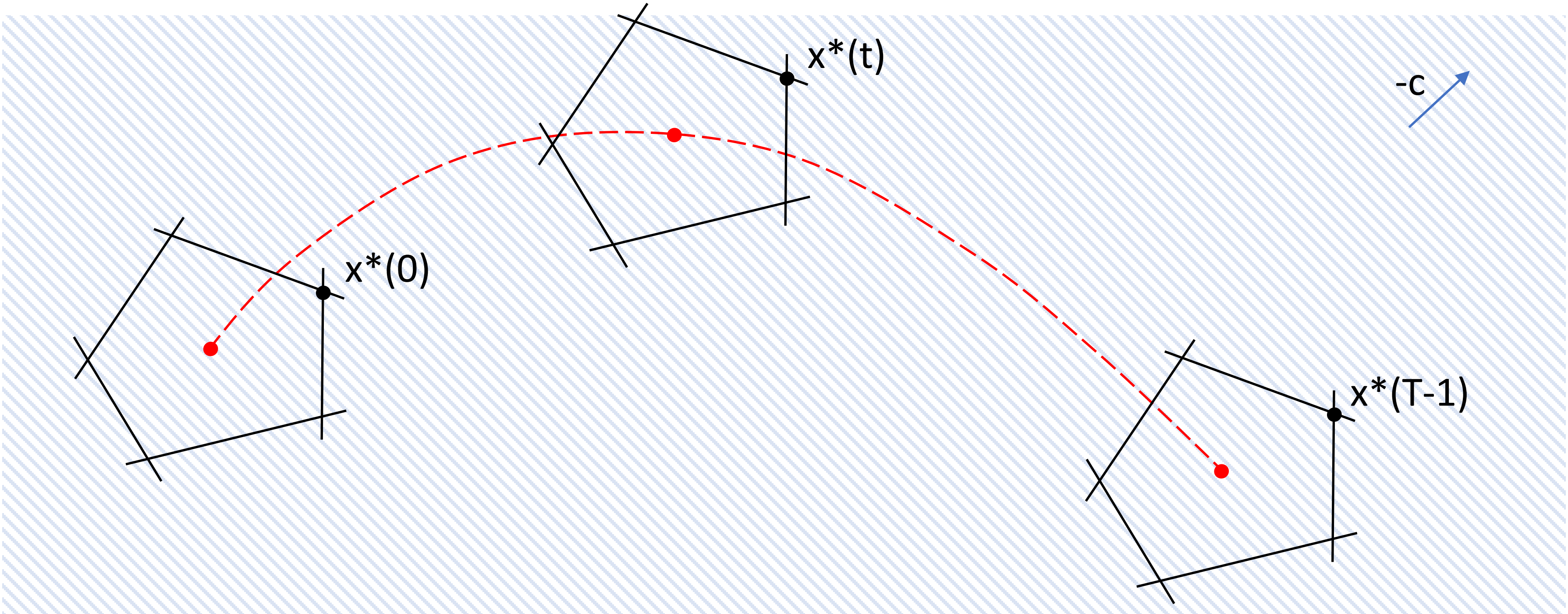
UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Departament de Teoria del Senyal
i Comunicacions

MOTIVATION

- Adaptive algorithms allow a system to adjust and improve its behavior over time in response to changing conditions. They provide better performance and reliability.
- For instance, Least Mean Square (LMS) is a classical gradient descent algorithm in which the weights of a model are adjusted based on the errors between predictions and the output.
- Nevertheless, LMS does not handle constraints. One choice could be to project the output of the gradient descent over the convex set, although convergence is not guaranteed.
- The goal of this project is to develop simple adaptative interior-point methods to assess the viability of adaptive schemes for convex optimization problems.

SETUP

- For $x \in R^2$, we consider a set of convex inequality and affine equality constraints.
- The dynamic environment is characterized by a continuous and smooth trajectory of the set.
  - The trajectory is defined by T steps (t=0...T-1).
  - The optimum at time t is x*(t).
- The example shows an LP.



GOALS

- Develop an adaptive **log-barrier** algorithm that can track the optimal x*(t).
- Develop an adaptive **primal-dual** algorithm that can track the optimal x*(t).
- Evaluate the algorithms for different scenarios:
  - Different constraint sets: different polyhedrons, other convex sets...
  - Different cost functions: linear, quadratic...
  - Different trajectories.
- Assess the benefits and drawbacks of both adaptive algorithms.
- Comments:
  - You can use off-the-self algorithms (i.e., Newton's method).
  - It is recommendable working with only two variables in order to visualize the tracking.
  - You are expected to show a visual proof of the tracking.

NOTES (I)

- How to generate **a dynamic set**:
  - Choose a center point within the set to parametrize it (e.g., analytical center).
  - Define the constraints with respect to the center.
  - Define a function for the trajectory in terms of the center point.
  - At every iteration: update the center and the constraint set.
- How to **plot results** (recommended, feel free to use others as well):
  - To generate a ground truth, solve the optimization problem for t=0...T-1.
  - Plot simultaneously the coordinates of the trajectory, the ground truth and the solution of your algorithm.

NOTES (II)

- **Experiment 1**:
  - Start with an LP and a linear trajectory (easiest example).
  - To avoid feasibility issues, make the trajectory so that the previous optimum is feasible after updating the set.
  - Test both, log-barrier and primal-dual algorithms.
- **Experiment 2**:
  - Same as Experiment 1 but update the trajectory in a direction such that the previous solution is not feasible in the current iteration.
  - Test both algorithms (log-barrier may not be possible).

NOTES (III)

- **Experiment 3**:
  - Update the LP with a quadratic cost in which the zero is found in the middle of the trajectory. For instance, f(x1,x2)=x1^2+x2^2 and S(x1,t;x2,t)=(x1-T/2+t,0). You can estimate the optimum by inspection.
  - Test both, log-barrier and primal-dual algorithms.
- **Experiment 4**:
  - Once the previous results are successful, try more complex cost functions, constraint sets and trajectories.

LOG-BARRIER PSEUDO-ALGORITHM

- Given: trajectory $S(t)$, initial constraint set $C(S(0))$, initial optimum $x^*(0)$.
- Repeat for $t=1...T-1$
  - Centering step: $x^*(t)$ by minimizing $k \cdot f0 + \phi$, subject to $Ax = b$, starting at $x^*(t-1)$
  - Update: (not needed)
  - Stopping criterion: (not needed, since optimization is required at every timestep)

- Notice we have substituted the parameter t in the original log-barrier to model the effect of the dynamic environment. Consequently, the problem needs a parameter k different from t.
- This is just a proposal, feel free to play around, since this may not work by itself.
- For the inner iterations in the centering step, we assume the dynamic fixed. We expect to perform few iterations if the previous optimal is within the quadratic convergence region.

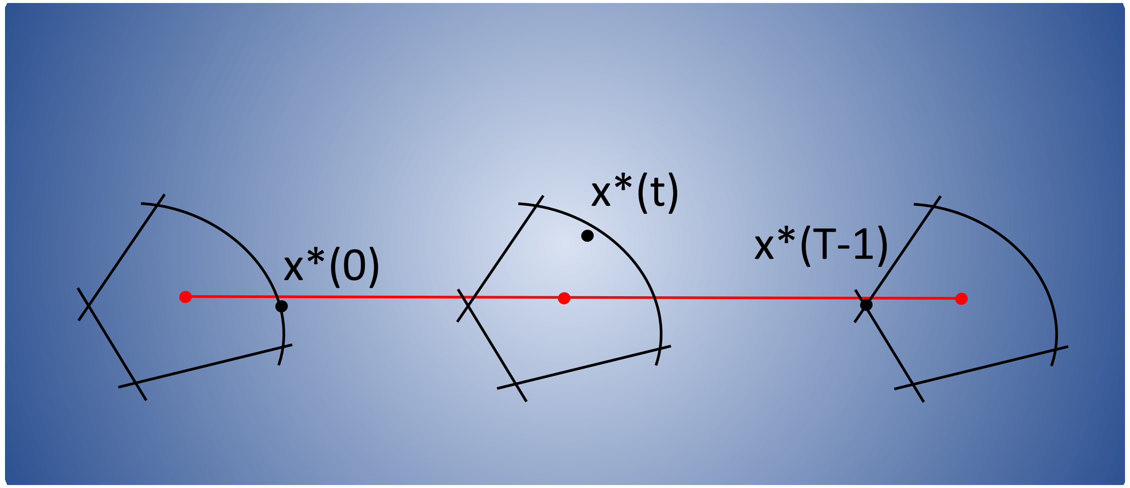
PRIMAL-DUAL PSEUDO-ALGORITHM

- Given: trajectory S(t), initial constraint set C(S(0)), initial optimum x*(0), $\lambda>0$, ν.
- Repeat for t=1...T-1
  - Determine k (surrogate duality gap).
  - Compute primal-dual search direction.
  - Line search and update: (not needed)
  - Stopping criterion: (not needed, since optimization is required at every timestep)

- Recall that the primal-dual method generates iterates that are not necessarily feasible. Does this generate an issue for an iterative algorithm that does not reach the exact optimal solution?

ALTERNATIVE EXPERIMENTS

- Experiment 3 with a new constraint set and different trajectory:



- Experiment 4: