Discrete Optimization

Summary WS 12/13

Linear Programming (1)

- Real-World Motivation
- Standard Forms
- Primal/Dual Simplex
- Duality

Linear Programming (2)

- potential exam questions (amongst others):
  - modelling of real world problem instance as (I)LP
  - solving a low-dimensional LP instance geometrically (bring ruler, pencil, eraser!)
  - describe primal/dual simplex
  - Transformations primal-dual

Linear Programming (3)

- very unlikely as a exam questions:
  - solve large-scale LP via explicitly solving sequence of systems of linear equations
  - exact syntax/call parameters of glpk
  - Intuitive interpretation of the dual (for new problems)

LP-based strategies for solving NP-hard optimization problems (1)

- Integer vs. fractional LP formulations
- Understanding how fractional/integral primal/dual solutions are useful in this context
- 3 main strategies:
  - dual fitting
  - rounding
  - primal-dual

LP-based strategies for solving NP-hard optimization problems (2)

- Possible exam questions (amongst others):
  - devise a rounding scheme for a given combinatorial optimization problem; prove a bound on the quality of its solution
  - recall analyses covered in class

LP-based strategies for solving NP-hard optimization problems (3)

- very unlikely as exam question:
  - ...

Network Flow (1)

- MaxFlow
  - Residual networks, augmenting paths ...
  - termination and correctness
  - faster, polynomial-time schemes
- MinCost Flow
  - only basic algorithm

Network Flow (2)

- possible exam question (amongst others):
  - model some real-world problem as a flow problem (with explanation)
  - solve small maxFlow/minCostFlow instances step-by-step
  - argue about correctness/termination

Network Flow (3)

- rather unlikely as exam question:
  - solve huge maxFlow/minCostFlow instance step-by-step
  - Nothing about Hall's theorem (exercise)

Non-LP-based Approximation Algorithms for NP-hard Problems (1)

- Vertex Cover
- Scheduling problems: independent tasks (PTAS), Precedence Constraint Scheduling
- (Fully)Polynomial-Time Approximation Schemes
- Dynamic Programming
  - Knapsack, FPTAS
  - Constrained Shortest Path, FPTAS

Non-LP-based Approximation Algorithms for NP-hard Problems (2)

- potential exam questions (amongst others):
  - solve CSP/Knapsack instance
  - explain covered algorithms/argue about quality of their solutions

Non-LP-based Approximation Algorithms for NP-hard Problems (3)

- rather unlikely as exam question:
  - solve huge CSP/Knapsack instance
  - No new inapproximability proofs
  - CSP part where one searches for the right scaling factor (the log log n thing)
  - No NP-hardness of CSP/Knapsack