20: Multiobjective EA

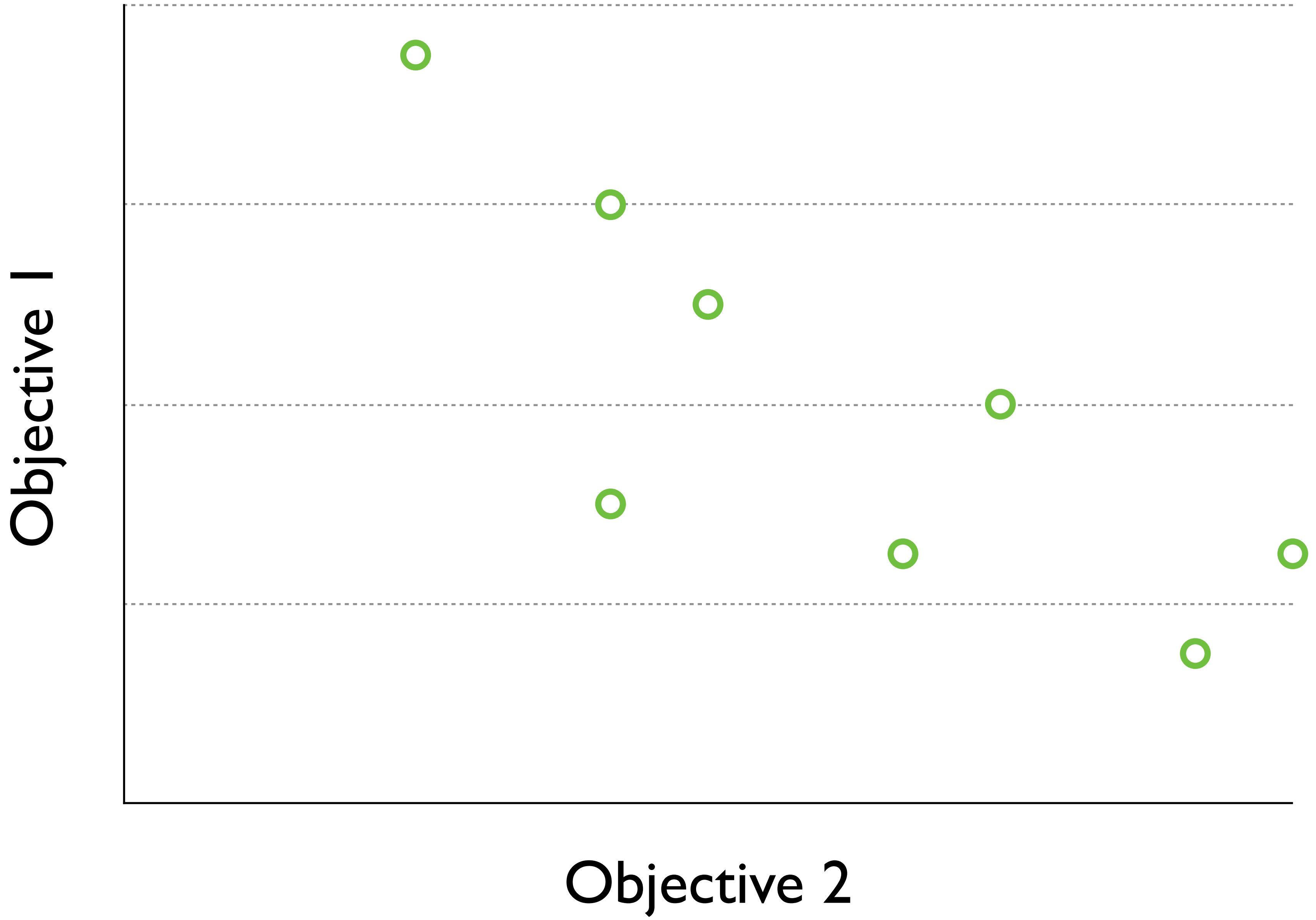
- Multi-objective optimization problems (MOP)
- Pareto optimality
- EC approaches to MOP
- Textbook chapter 12

Prof. Ting Hu, School of Computing, Queen's University

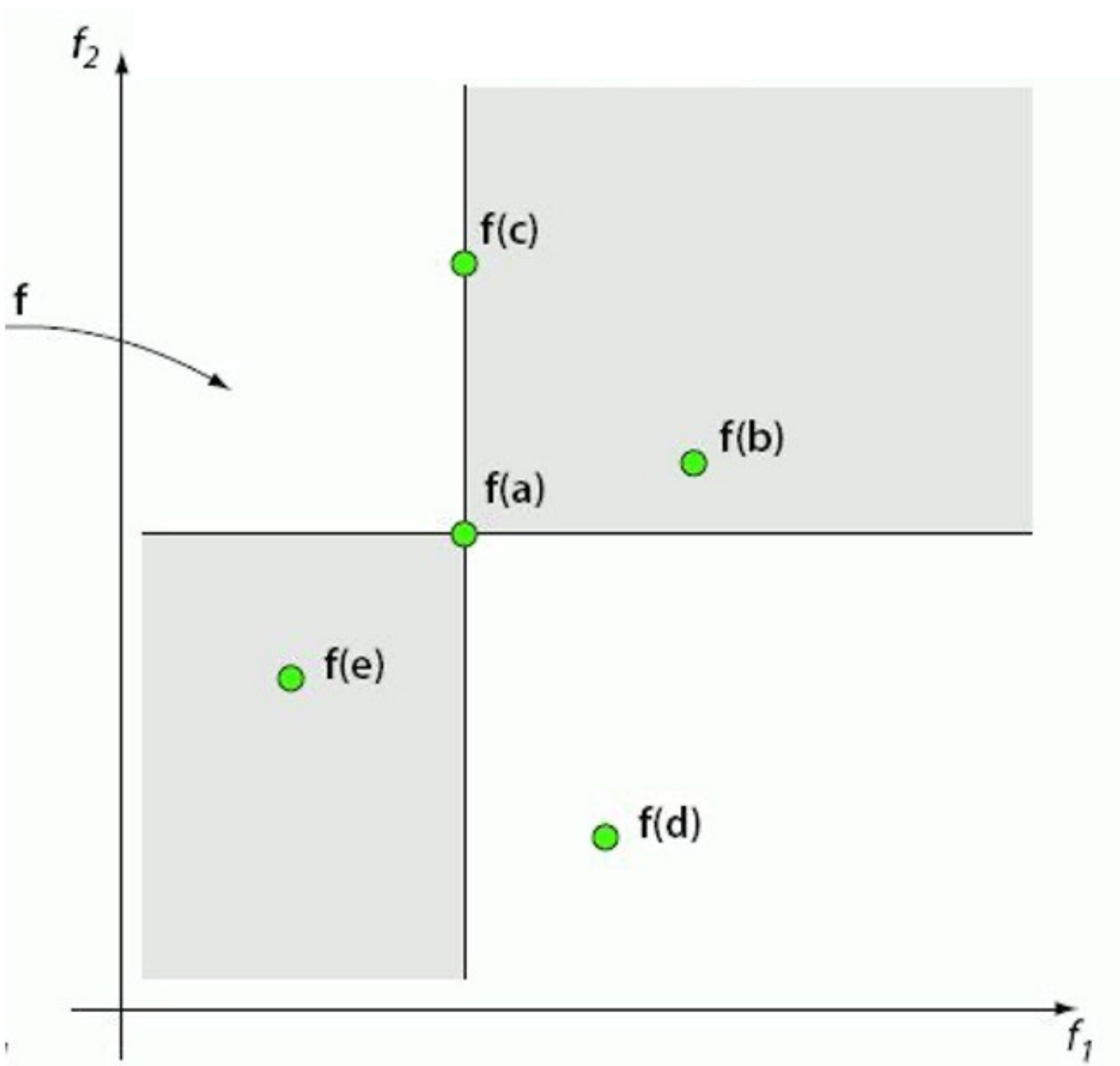
Multi-objective problems (MOPs)

- The presence of a number of possibly conflicting objectives
  - buying a house
  - designing an electric car
- Two steps
  - find a set of good solutions
  - choice of best for particular applications

Objective space

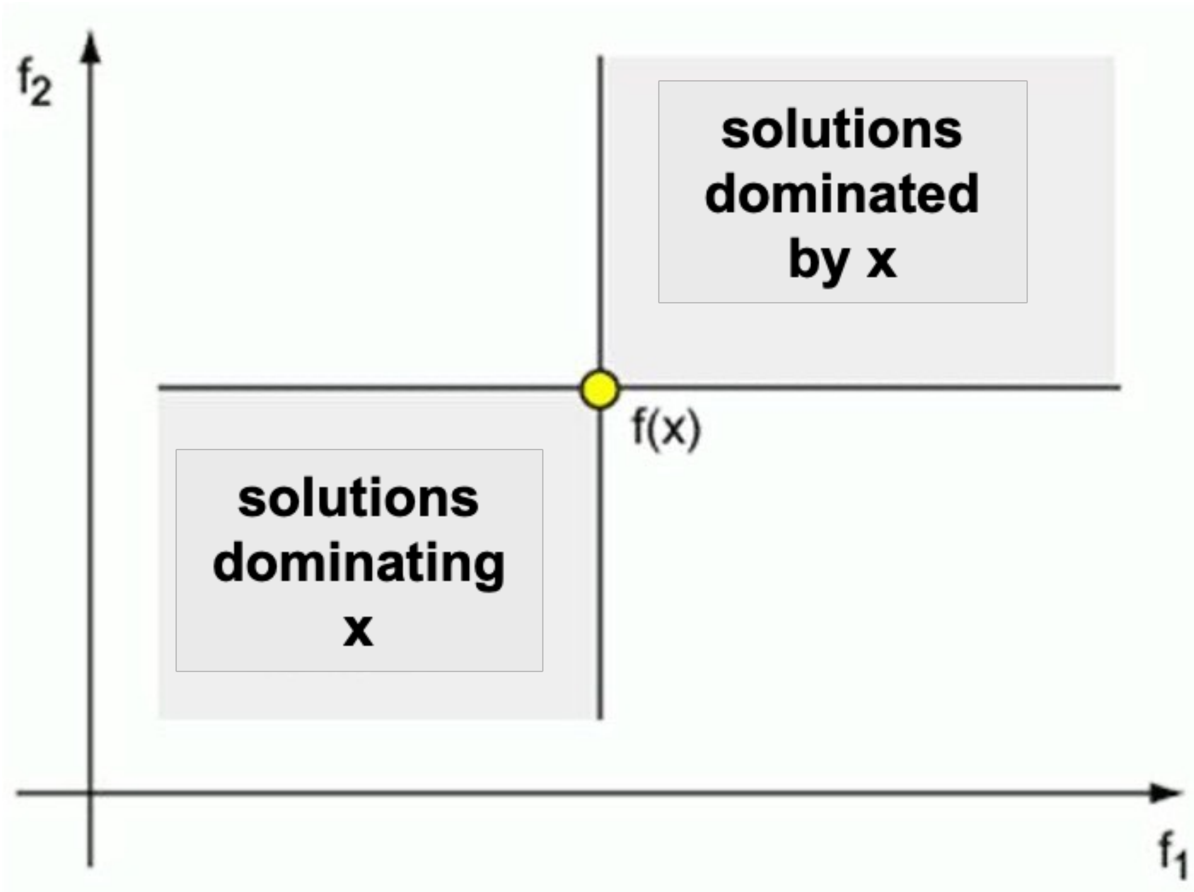


Comparing solutions



- Optimization task
  - minimize both f_1 and f_2
- Compare the solutions
  - a vs. b
  - a vs. c
  - a vs. e
  - a vs. d

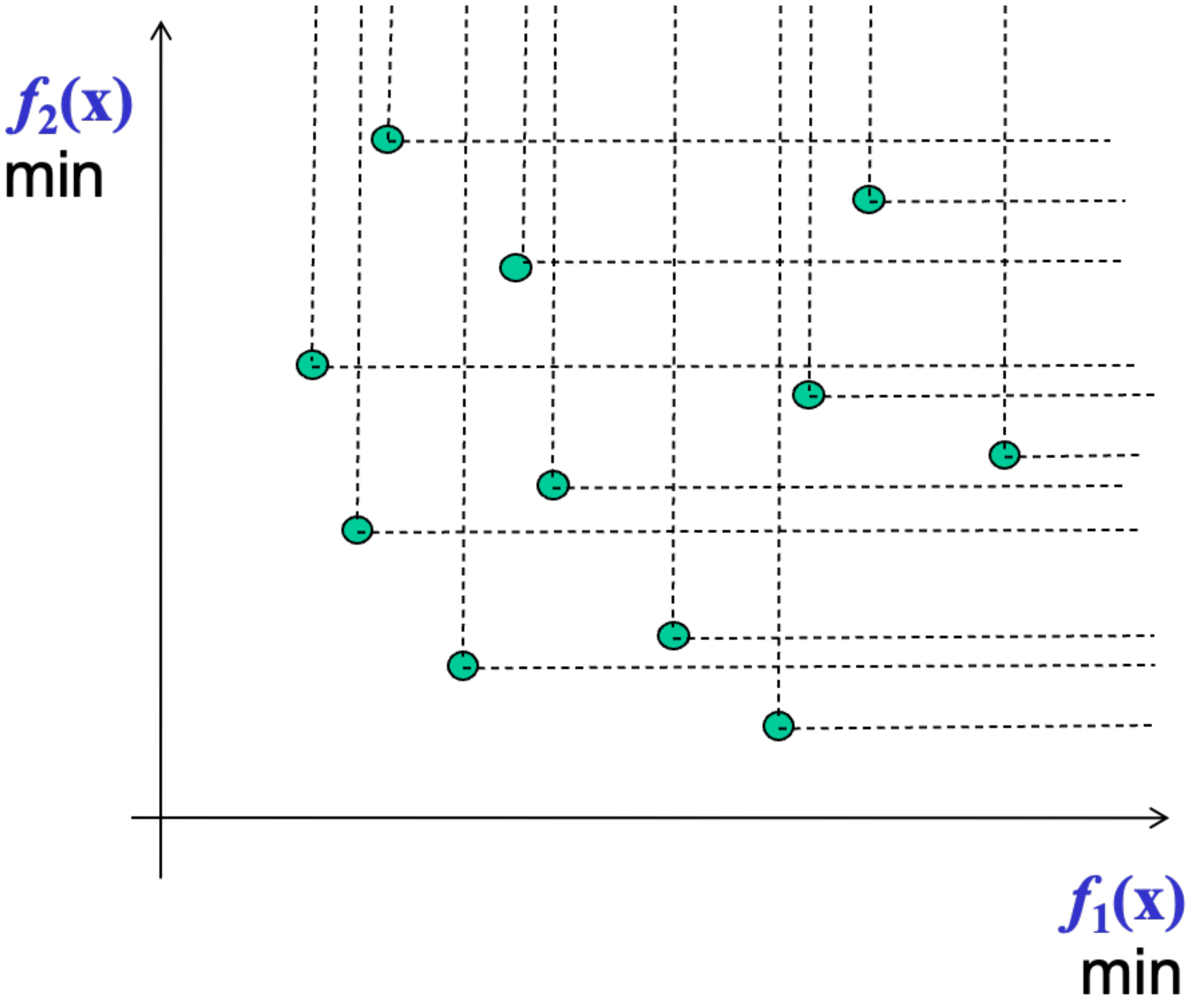
Dominance relation



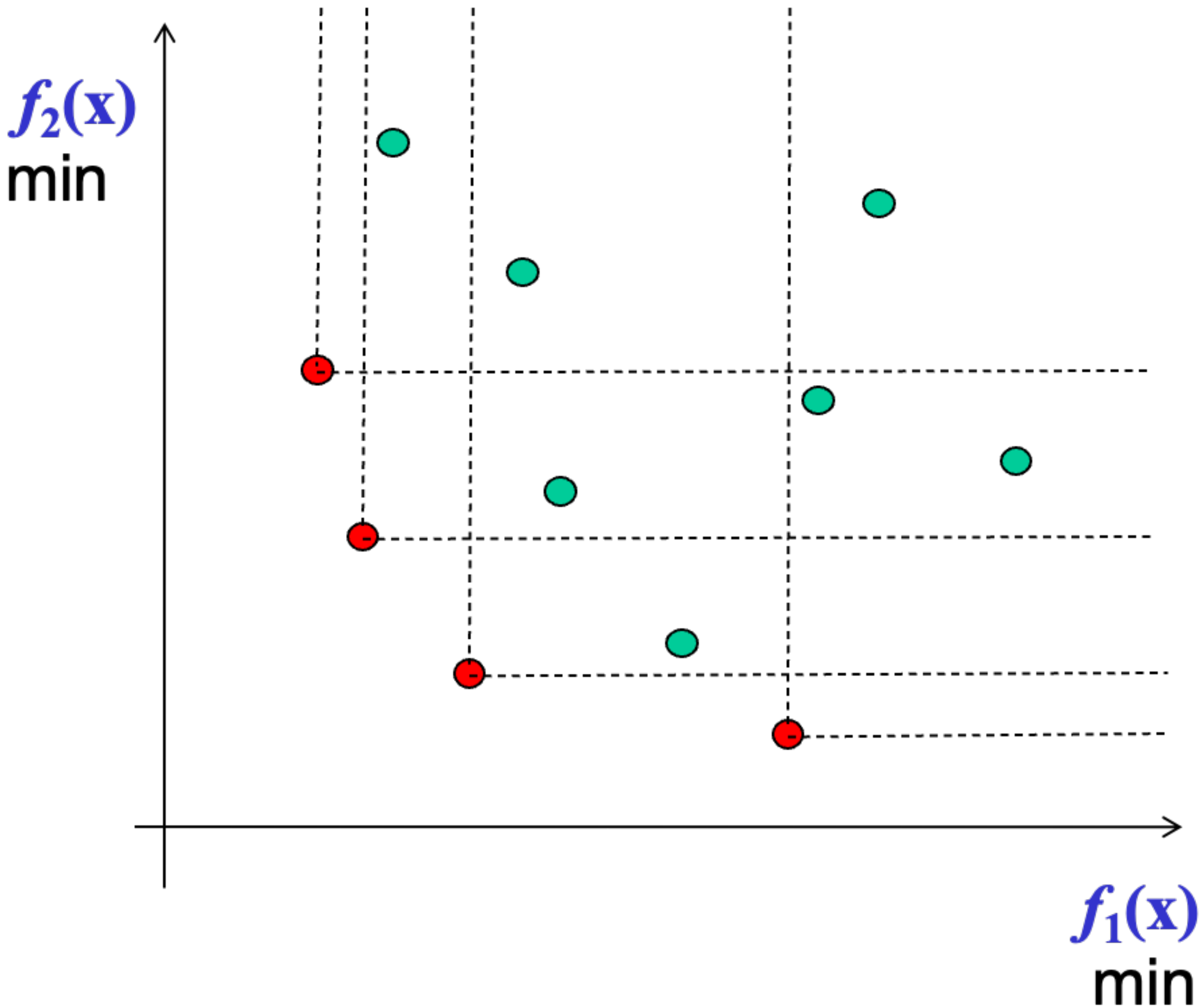
- Solution x dominates solution y
  - x is not worse than y in all objectives
  - x is better than y in at least one objective

Pareto optimality

- Solution x is non-dominated among a set of solutions Q: no solution from Q dominates x
- A set of non-dominated solutions from the entire feasible solution space: the Pareto-optimal set, its members Pareto-optimal solutions
- Pareto-optimal front or Pareto front: a visualization of the Pareto-optimal set in the objective space

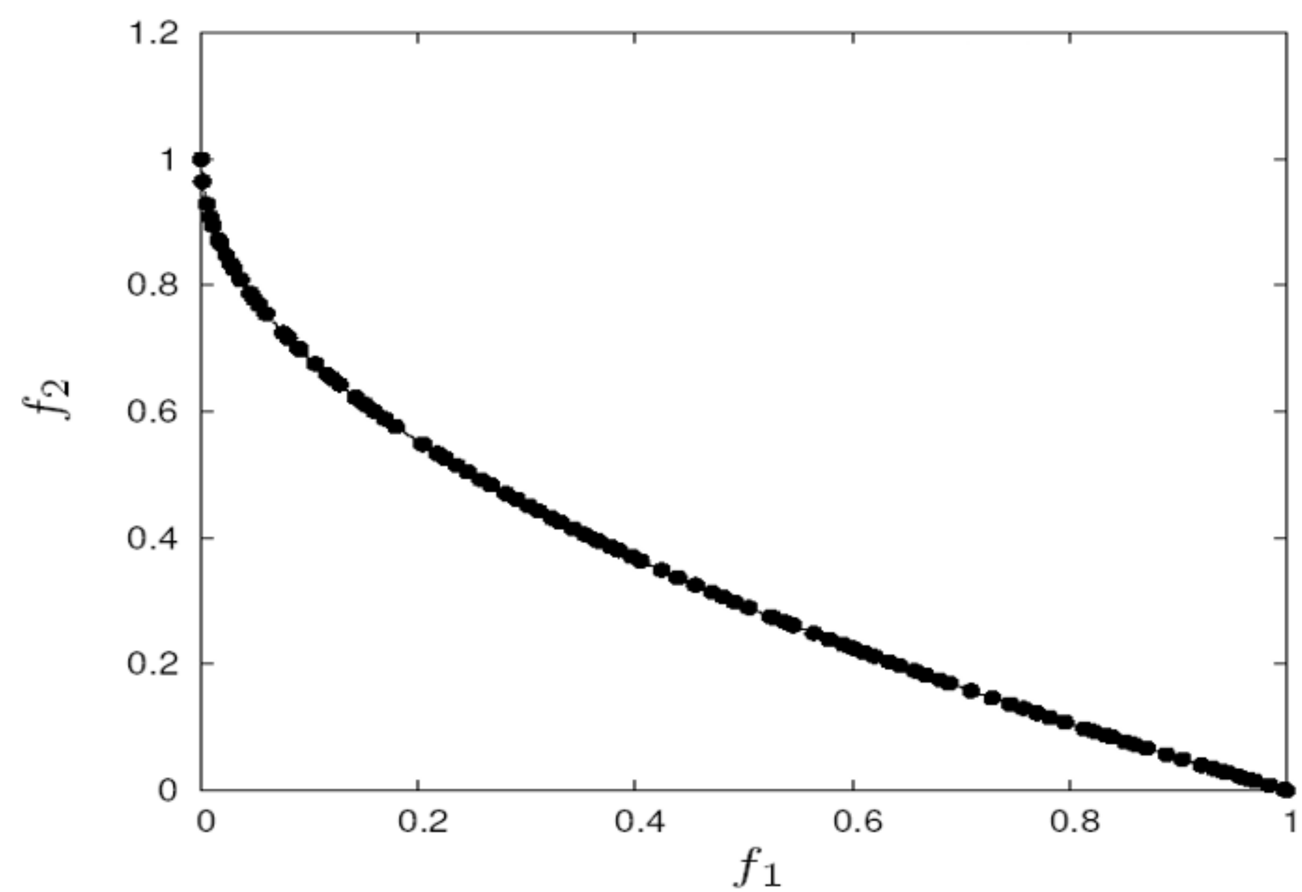


$f_2(\mathbf{x})$
min
$f_1(\mathbf{x})$
min


$f_2(\mathbf{x})$
min
$f_1(\mathbf{x})$
min

Goal of multi-objective optimization

- Find a set of non-dominated solutions (approximation set)
  - convergence (as close as possible to the Pareto-optimal front)
  - diversity (spread, distribution)



Single- vs. multi-objective optimization

Characteristics	Single-objective	Multi-objective
# of objectives	1	>1
Comparison of solutions	x is better than y	x dominates y
Result	one or more equally good solution(s)	Pareto-optimal set
Algorithm goals	convergence	convergence, diversity

Approaches to multi-objective optimization

- Preference-based approaches:
  - need good understanding on importance of different objectives
  - combine multiple objective optimizers
  - assign weights to multiple objectives
  - optimize weighted sum

$$F(\mathrm{X}) = \sum_{m=1}^{M} w_m f_m(\mathrm{X}), \quad w_m \in [0,1], \quad \sum_{m=1}^{M} w_m = 1$$

Could be better if

- Solve the multi-objective optimization problem using one method that
    - finds multiple trade-off solutions
    - later for users to decide on a particular trade-off solution
    - with higher-level domain knowledge

Multi-objective evolutionary algorithms

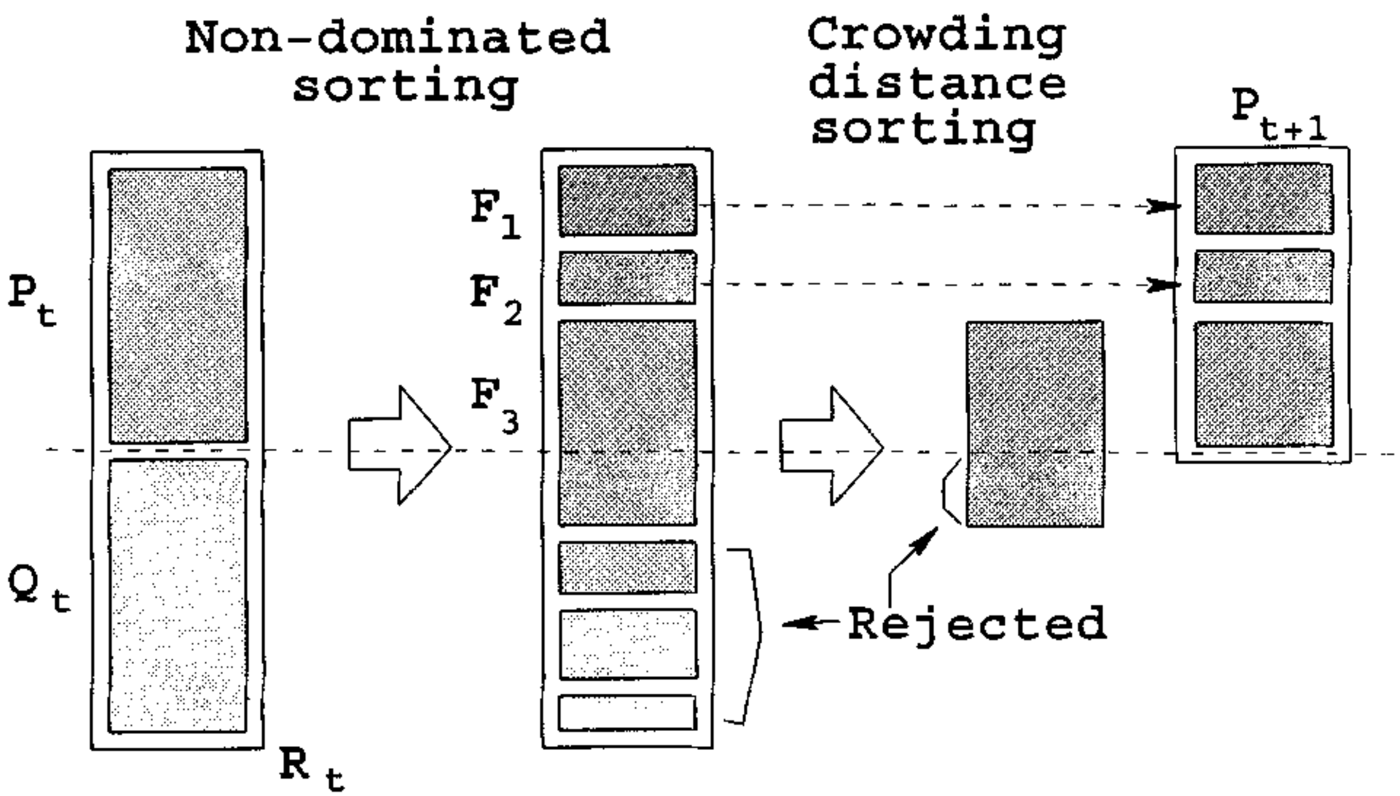
- Population-based method
- No need to assume the weights
- No assumption about the shape of Pareto-front (convex, continuous, etc.)
- Can return a set of trade-off solutions (approximation set) in a single run
- Allows for users to pick the more suitable trade-off solutions
    - preference may change for different instances
    - for different users
    - under different circumstances

Multi-objective EAs

- Fitness function
  - usually based on dominance
  - e.g., in MOGA, number of individuals of the current population one dominates
- Preservation of diversity
  - techniques discussed for multi-modal problems, crowding, fitness sharing, etc.
  - e.g., in NPGA, tournament selection on dominance and then similarity
- Remember all the non-dominated solutions ever seen
  - usually use elitism or an archive

Non-dominated sorting genetic algorithm (NSGA II)

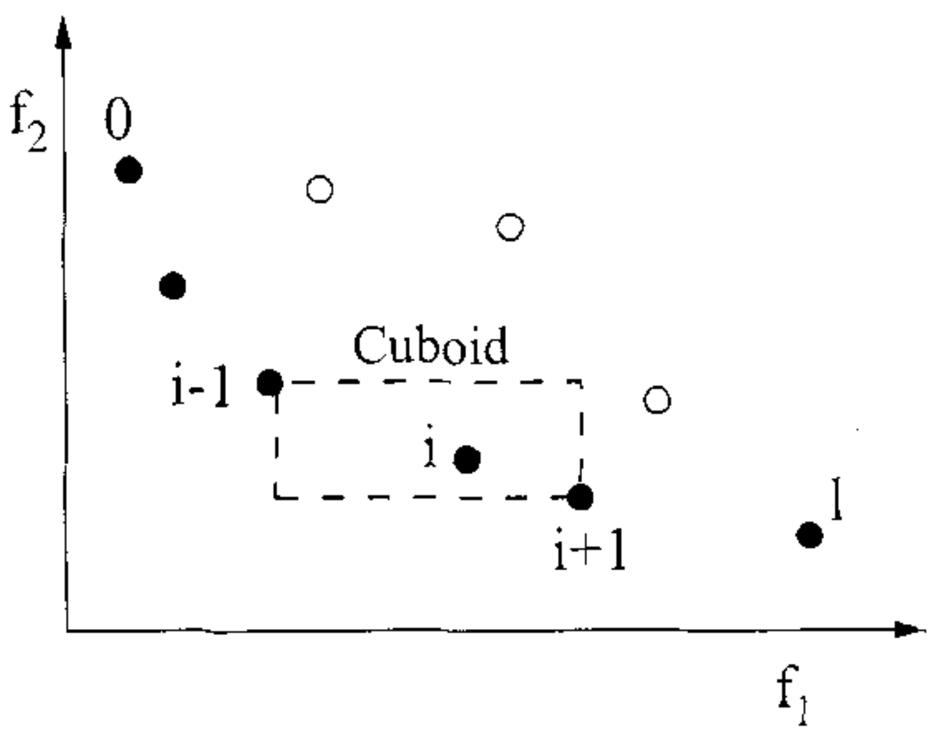
- Proposed by Deb et al. in 1994 (NSGA) and 2002 (NSGA II)
- Divide a current population into fronts of equal domination
- $(\mu + \lambda)$ survivor selection (elitism)



Deb et al., *IEEE Transaction on Evolutionary Computation*, 6(2):182-197, 2002

NSGA II crowding

- Secondary selection criterium - preserve diversity
- Crowding distance - the average side length of the cuboid defined by its nearest neighbors in the same front



Deb et al., *IEEE Transaction on Evolutionary Computation*, 6(2):182-197, 2002