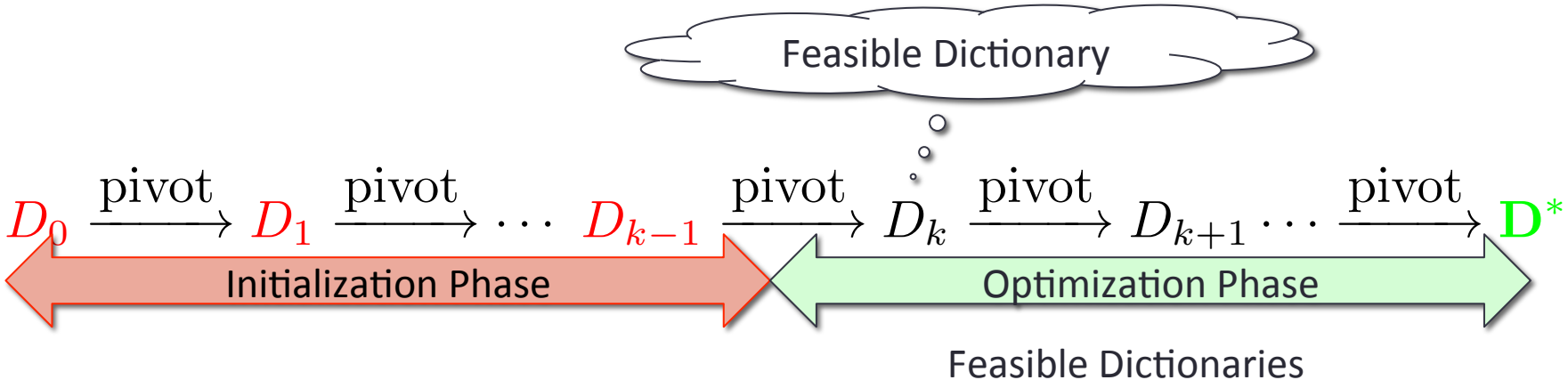
CYCLING IN SIMPLEX

Does simplex always terminate?

Simplex Overview



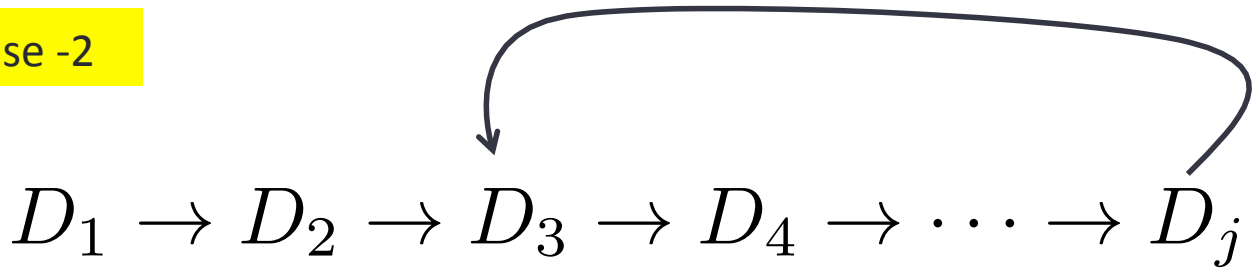
Termination of Simplex

Case -1

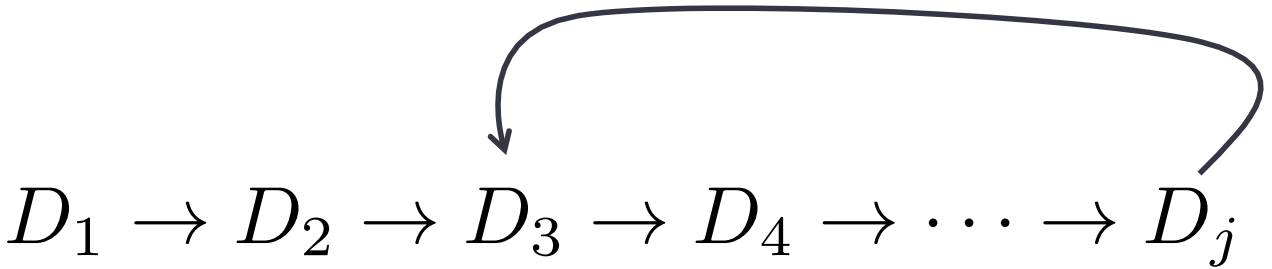
Infinitely many dictionaries?

$$D_1 \to D_2 \to D_3 \to D_4 \to \cdots \to D_j \to \cdots$$

Case -2



Cycling in Simplex



Only possible case for non-termination of Simplex.
Is this possible?

Cycling and Degeneracy

$$D_1 \to D_2 \to D_3 \to D_4 \to \cdots D_i \to \cdots \to D_j$$

All repeating dictionaries $D_i, \ldots, D_j$ are *degenerate*

Cycling in Simplex

- Depends on heuristic for choosing entering/leaving variables.
  - Lots of examples
  - See on-line or consult Chvatal/Vanderbei book.

Anti-Cycling Rule

- Can the choice of entering or leaving variable avoid cycling and guarantee termination of Simplex?

- Good news ☺
  - Bland's Rule is anti-cycling.

- Recall Bland's Rule:
  - If multiple choices for entering, choose least index.
  - If multiple choices for leaving, once again choose least index.

Practical Considerations

- Stalling

Analysis of Bland's Rule vs. Other Heuristics

- Classic Paper by Avis + Chvatal (1974)
- On small randomly generated problems:
  - Bland's rule performs worse than other heuristics such as largest objective coefficient and greedy.
  - Stalling seems to be made worse by Bland's rule in many situations?
- Suggestion:
  1. Cycling never seems to happen in practice.
  2. Use Bland's rule selectively.

Example of Mixed Heuristic

- Use largest objective coefficient rule for selecting entering variable.

- If last K dictionaries are all degenerate
  - apply Bland's rule continuously until final or non-degenerate dictionary.