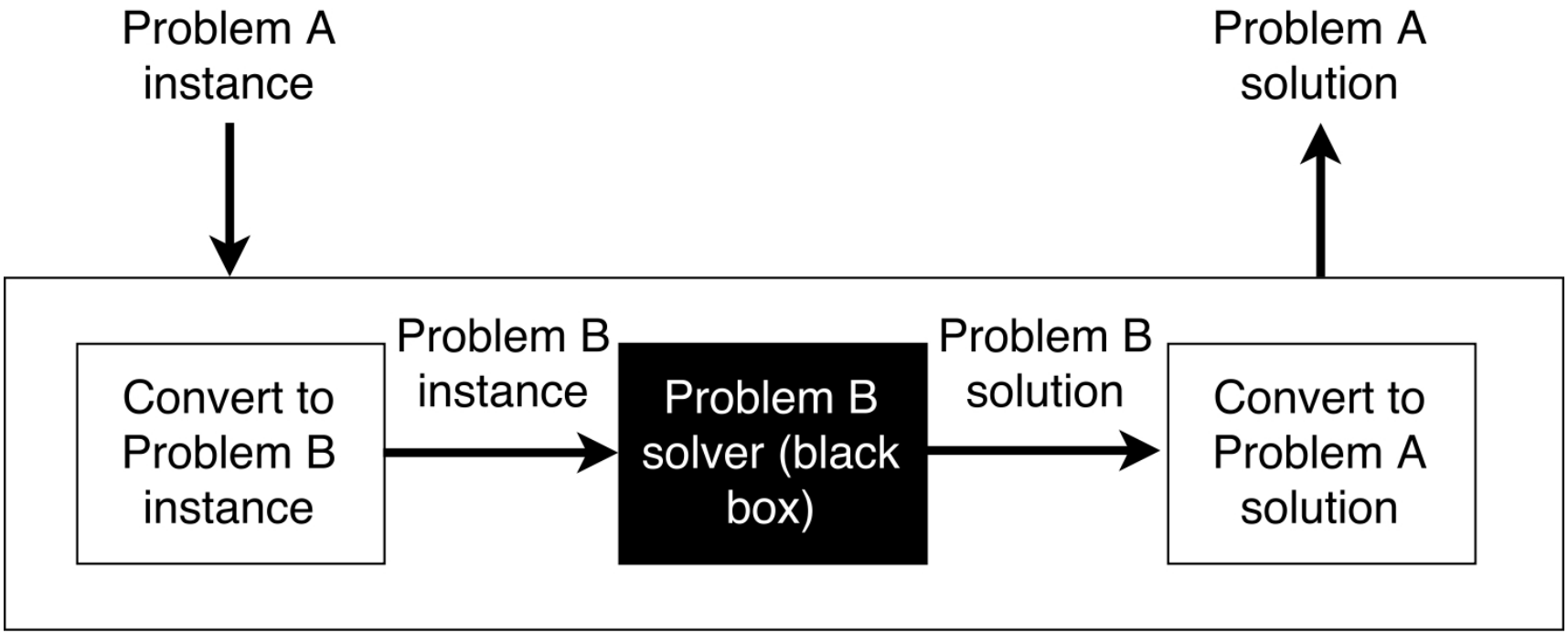
Reductions

CPSC 320 2024W1

Context

- You encounter a new problem **A**
  - You don’t have an algorithm to solve it
  - But it seems similar to a problem **B** that you do know how to solve
  - … so, you might use your solver for **B** rather than writing an entirely new solver for **A**
- Definition: an *instance* of a problem is a valid input

How a reduction works



How a reduction works (continued)

You need to:

- Show how to transform an arbitrary instance $\mathbf{I_A}$ of **A** into an instance $\mathbf{I_B}$ of **B**
- Show how to transform a solution $\mathbf{S_B}$ of $\mathbf{I_B}$ into a solution $\mathbf{S_A}$ of $\mathbf{I_A}$
- Prove that $\mathbf{S_A}$ is a correct solution to $\mathbf{I_A}$, assuming that $\mathbf{S_B}$ is a correct solution to $\mathbf{I_B}$

The total running time is:

- The sum of the two transformations
- Plus the time to solve $\mathbf{I_B}$

Reduction example

- **A:** Given a list of numbers $[X_1, X_2, \ldots, X_n]$, find the smallest gap between any two of them
- **B:** Sorting a list of values
- **Reduction:**
  - Given an instance I_A of **A**, let $I_B = I_A$
  - Sort the list I_B to get a list $[Y_1, Y_2, \ldots, Y_n]$
  - Return $\min_{i=1,2,\ldots,n-1}\{Y_{i+1} - Y_i\}$

Reduction example and clicker question

- **A:** Given a list of numbers **[X_1, X_2,..., X_n]**, determine if there are any duplicates in the list
- **B:** Given a list of numbers **[X_1, X_2,..., X_n]**, find the smallest gap between any two of them
- **Reduction:**
  - Given an instance I_A of **A**, let $I_B = I_A$
  - Solve I_B to get the value **smallestGap**
  - Return True to A if  ; else return False

Reduction example 3

- **A:** Given a SAT instance where all clauses have length 2, is there a way to assign truth values to variables such that all clauses evaluate to True? (AKA, 2-SAT)
  - E.g., $(x \vee \overline{y}) \wedge (y \vee z)$
- **B:** Given a SAT instance where all clauses have length 3, is there a way to assign truth values to variables such that all clauses evaluate to True? (3-SAT)
- **Reduction:**
  - Given an instance $\mathbf{I_A}$ of **A**, repeat the last literal in each clause and "OR" it to the previous literal to obtain $\mathbf{I_B}$, e.g., $(x \vee \overline{y} \vee \overline{y}) \wedge (y \vee z \vee z)$
  - Solve $\mathbf{I_B}$ to obtain a truth assignment $\mathbf{S_B}$ (if it exists, or the value NO if $\mathbf{I_B}$ is not satisfiable
  - $\mathbf{S_A = S_B}$