# Ch. 5 Computation

# 1 Programs

## 1.1 Logical Instructions

Define $\mathcal{L}$; an ordered set of logical operations $s_i$

$$\mathcal{L} := \{s_1, s_2, ..., s_N\}$$

## 1.2 Memory

Define Memory $\mathcal{M}$; an ordered set of elements, magnitudes, or sets

$$\mathcal{M} := \{b_1, b_2, ..., b_M\}$$

## 1.3 State |

Define state; the memory utilized to perform program P

$$P := \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M\} =$$
$$\{s_1, s_2, ..., s_N, b_1, b_2, ..., b_M\}$$

## 1.4 Boolean Programs

Define a boolean program; boolean programs can represent functions with inputs $x_i$, input set C, and boolean output $y_o$

$$X = \{x_1, ..., x_n, C\}; \;\; C = \{u_1, u_2, ..., u_c\}$$
$$P = P[X] := \{s_1, s_2, ..., s_N \mid b_1, b_2, ..., b_M, y_o\} =$$
$$P[X] \rightarrow y_o \in \{\mathbb{T}, \mathbb{F}\}$$

## 1.5 Void Programs

Define a void program; a program with inputs $x_i$, input set C, and no output

$$X = \{x_1, ..., x_n, C\}$$
$$P = P[X] := \{s_1, s_2, ..., s_N \mid b_1, b_2, ..., b_M\}$$

### 1.6 Numerical Programs

Define a numerical program; a program with inputs $x_i$, input set C, and real, rational output $y_o$

$$X = \{x_1, ..., x_n, C\}$$
$$P = P[X] := \{s_1, s_2, ..., s_N \mid b_1, b_2, ..., b_M, y_o\} =$$
$$P[X] \to y_o \in \mathbb{Q} \;\; y_o \geqslant 0$$

### 1.7 System Programs

Define a system program; a program with inputs $x_i$, input set C, and real, output set $Y_o$

$$X = \{x_1, ..., x_n, C\}$$
$$P = P[X] := \{s_1, s_2, ..., s_N \mid b_1, b_2, ..., b_M, Y_o\} =$$
$$P[X] \to Y_o = \{y_1, y_2, ..., y_K\}$$

### 1.8 Mathematical Programs

Define a mathematical program; a program with inputs $x_i$, input set C and numerical output $y_o$

$$X = \{x_1, ..., x_n, C\}$$
$$P = P[X] := \{s_1, s_2, ..., s_N \mid b_1, b_2, ..., b_M, y_o\} =$$
$$P[X] \to y_o \in \mathbb{Q}$$

# 2 No-op ;

## 2.1 Definition

$$; := \varnothing$$

## 2.2 Property of No-op

No-op can be inserted into any set with equality

$$S = \{s_1, s_2, ..., s_N\}$$

$$S_; = insert[S, ;, i]$$

$$S_; = S_1 \ \ \forall i$$

$$|S_;| = |S| \ \ \forall i$$

## 2.3 Proof

by definition of magnitude of null = 0 with Set And

# 3 Decision Problems

## 3.1 Definition

Define decision problem; a function with inputs $x_i$ and boolean output "answer" $a_o$

$$X_i = \{x_1, ..., x_n, C\}$$
$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

# 4 General Solutions

## 4.1 Definition

Program P is a general solution $s^+$ to decision problem D if

1. P outputs answer $a_o$ for all inputs $X_i$ $\forall i$
and
2. $s^+[X_i]$ is a subset of $s^+[\hat{X}_i]$

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_n, x_{n+1}, C\}$$
$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$
$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \to y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$
$$P[X_i] = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\}$$
$$s^+ = P[X_i] = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} \quad \forall X_i$$

### 4.1.1 Property of No-op ;

No-op ; can be added to any solution $S_i$ without modifying the output $y_o$

$$s^+ = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$\hat{s}^+ \to \hat{y}_o = insert[s^+, ;, k]$$
$$\hat{y}_o = y_o \;\; \forall k$$

## 4.2 Definition of $S^+$

Define $S^+$; the set of solutions to decision problem D

$$X_i = \{x_1, ..., x_n, C\}; \quad \hat{X}_i = \{x_1, ..., x_n, x_{n+1}, C\}$$
$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$
$$s_j^+ = s_j^+[X_i] := P :$$
$$(P[X_i] \to y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$
$$S^+ := \{s_j^+, ...\} \quad \forall j$$

### 4.3 Definition of Solvable

Define solvable

$$X_i = \{x_1, ..., x_n, C\}; \quad \hat{X}_i = \{x_1, ..., x_n, x_{n+1}, C\}$$
$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$
$$solvable := solvable[D] \to b_o \in \{\mathbb{T}, \mathbb{F}\} =$$
$$\exists P : (P[X_i] \to y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

## 5 The set of all Decision Problems $\mathbb{D}$

### 5.1 Definition

Define the set of decision problems $\mathbb{D}$

$$X_i = \{x_1, ..., x_n, C\}$$
$$D_j := f_j[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$
$$\mathbb{D} := \{D_j, ...\} \quad \forall j$$

## 6 Instruction and Memory Notation

Define $\mathcal{L}$ a set of logical operations
Define $\mathcal{M}$ a set of memory elements, magnitudes, and sets

$$X_i = \{x_1, ..., x_n, C\};$$
$$P[X_i] \to y_o = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$\mathcal{L} := \{s_1, s_2, ..., s_{O_T[n]}\}$$
$$\mathcal{M} := \{b_1, b_2, ..., b_{O_S[n]}\}$$
$$P[X_i] = \{\mathcal{L}, \mathcal{M}, y_o\}$$

# 7 Complexity

## 7.1 Time Complexity of a Decision Problem $O_T[n]$

Define Time Complexity $O_T[n]$ of solution $s^+$ to Decision Problem $D$ as the total number of logical operations

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$

$$D := f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

$$s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$\begin{aligned} s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} &= \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} \\ &= \{\mathcal{L}, \mathcal{M}, y_o\} \end{aligned}$$

$$O_T[n] := |\mathcal{L}| = N$$

## 7.2 Space Complexity $O_S[n]$

Define Space Complexity $O_S[n]$ of solution $s^+$ to Decision Problem $D$ as the total number of memory elements

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$

$$D := f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

$$s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$\begin{aligned} s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} &= \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} \\ &= \{\mathcal{L}, \mathcal{M}, y_o\} \end{aligned}$$

$$O_S[n] := |\mathcal{M}| + |y_o|^* = M + 1$$

*It is convention to reserve one memory element for output $y_o$.
Void programs do not require the $y_o$ memory element for output

# 8 Definition of Complexity

Define Complexity $O[n]$ as a vector of dimension Y

$$\mathbf{O}[n] := \ \ < O_T[n], O_S[n], O_3[n], O_4[n] ..., O_V[n] >$$

# 9 Total Complexity

$$O[n] := O_T[n] + O_S[n] + \sum_{i=3}^{V} O_i[n]$$

# 10 Simple Computational Complexity

The remainder of this chapter assumes simple computational complexity of dimension 2

## 10.1 Definition

Define simple computational complexity of dimension 2

$$\mathbf{O}[n] := \; < O_T[n], O_S[n] >$$

## 10.2 Time Complexity

Restate definition of Time Complexity $O_T[n]$ of solution $s^+$

$$s^+ = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$O_T[n] := |\mathcal{L}| = N$$

## 10.3 Space Complexity

Restate definition of Time Complexity $O_S[n]$ of solution $s^+$

$$s^+ = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$O_S[n] := |\mathcal{M}| + |y_o| = M + 1$$

## 10.4 Total Complexity

$$O[n] := O_T[n] + O_S[n]$$

$$= |\mathcal{L}| + |\mathcal{M}| + |y_o| = N + M + 1$$

## 10.5 $O_S[n] \neq 0$

### 10.5.1 Proof

By definition of decision problem; Proof by contradiction; $y_o$ must be set to TF by definition; Suppose yo = 0; then yo is empty set; contradicts definition of D

## 10.6 $O_T[n] \neq 0$

### 10.6.1 Proof

By definition of decision problem; Proof by contradiction; $y_o$ must be set to TF by definition; Suppose |L| = 0; yo <- TF cap L is null by definition of empty set; implies yo emptyset (doesnt exist)

## 10.7 $\mathbf{O[n]} = O_T[n] + O_S[n] \neq \mathbf{0}$

### 10.7.1 Proof

## 10.8 $O[n] > O_T[n]$

### 10.8.1 Proof

## 10.9 $O[n] > O_S[n]$

### 10.9.1 Proof

## 10.10 $O[n+1] \geqslant O[n]$

## 11 Polynomial Complexity

### 11.1 Definition

Decision problem $D$ with solution $s^+$ has polynomial total complexity $O[n]$ if

$$\exists K, C, \lambda_1 ... \lambda_K \quad :$$
$$O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

### 11.2 Polynomial Problems

Define $\mathbb{P}$, the set of Decision Problems that can be solved with Polynomial Complexity

$$\mathbb{P} := \{D_1, D_2, ...\} :$$
$$\exists K, C, \lambda_1 ... \lambda_K :$$
$$O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n, D_i \in \mathbb{P}$$

### 11.3 Polynomial Order of Complexity

Solution $s^+$ with total complexity $O[n]$ is said to be of order $n^K$

$$O[n] \sim n^K$$
$$O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

### 11.4 Property of Polynomial Complexity 1

$$lim_{n \to \infty} \frac{O[n+1]}{O[n]} = 1$$

#### 11.4.1 Proof WIP

FIX!!! Show there exists no constant satisfying the decreasing limit condition

$$O[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C$$
$$O[n+1] < (\lambda_K (n+1))^K + (\lambda_{K-1} (n+1))^{K-1} ... + \lambda_1 (n+1) + C$$
$$O[n] \sim (\lambda n)^K; \;\; O[n+1] \sim (\lambda n)^K$$
$$lim_{n \to \infty} \frac{(\lambda n)^K}{(\lambda n)^K} = 1$$

## 11.5 Property of Polynomial Complexity 2

$$\exists K, C, \lambda_1, ..., \lambda_K :$$
$$(O[n+1] - O[n]) = f_{n+1}[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

### 11.5.1 Proof FIX!!!

$$O[n+1] < (\lambda_K (n+1))^K + (\lambda_{K-1}(n+1))^{K-1} ... + \lambda_1 (n+1) + C$$
$$O[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C$$

## 11.6 Total Polynomial Complexity Implies Time bounded Polynomial Complexity

$$D \in \mathbb{P} \Longrightarrow O_T[n] < ...$$

### 11.6.1 Proof FIX!!!

$$O[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n$$
$$O[n] := O_T[n] + O_S[n]; \;\; O_T[n] < O[n]$$
$$\therefore O_T[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n$$

## 11.7 Total Polynomial Complexity Implies Space bounded Polynomial Complexity

$$D \in \mathbb{P} \Longrightarrow O_S[n] < ...$$

### 11.7.1 Proof FIX!!!

$$O[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n$$
$$O[n] := O_T[n] + O_S[n]; \;\; O_S[n] < O[n]$$
$$\therefore O_S[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n$$

## 11.8 Total Polynomial Complexity iff Time and Space bounded by Polynomial Complexity

Use limit definition

## 11.9 Order of Complexity

ERROR in second condition
Total Complexity is said to be on the order of $K_{max}$

$$O[n] \sim K_{max}$$

$$K_{max} := K :$$

$$O[n] < (\lambda_{K_{max}} n)^{K_{max}} + (\lambda_{K_{max}-1} n)^{K_{max}-1} ... + \lambda_1 n + C \quad \forall n$$

$$\nexists O[n] < (\lambda_{\hat{K}_{max}} n)^{\hat{K}_{max}} + (\lambda_{\hat{K}_{max}-1} n)^{\hat{K}_{max}-1} ... + \lambda_1 n + C \quad \forall n, \hat{K} < K_{max}$$

## 11.10 Theorem Either OT or OS is on the order of Oopt

Proof by contradiction

# 12 Polynomial Time Complexity

## 12.1 Definition

Decision problem $D$ with (optimal) Time Complexity $O_T[n]$ is bounded by polynomial time complexity if

$$\exists K, C, \lambda_1 ... \lambda_K \quad :$$
$$O_T[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

## 12.2 Polynomial Time Solutions

Define $\mathbb{S}^+_{time}$, the set of solutions that can be solved with polynomial time complexity

$$\mathbb{S}^+_{time} := \{s^+_1, s^+_2, ...\} :$$
$$\exists K, C, \lambda_1 ... \lambda_K :$$
$$O_T[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n, s_i \in \mathbb{S}^+_{time}$$

## 12.3 Property of Polynomial Time Complexity 1

$$lim_{n \to \infty} \frac{O_T[n+1]}{O_T[n]} = 1$$

### 12.3.1 Proof

## 12.4 Property of Polynomial Time Complexity 2

$$\exists K, C, \lambda_1, ..., \lambda_K :$$
$$(O_T[n+1] - O_T[n]) < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n$$

### 12.4.1 Proof

## 12.5 Order of Complexity

Time complexity $O_T[n]$ is said to be on the order of $K_{max}$

$$O_T[n] < (\lambda_{K_{max}} n)^{K_{max}} + (\lambda_{K_{max}-1} n)^{K_{max}-1} ... + \lambda_1 n + C$$
$$O_T[n] \sim K_{max}$$

## 12.6 Proof of the existence of $O_{T_{opt}}$

# 13 Polynomial Space Complexity

## 13.1 Defintion

Decision problem $D$ with (optimal) Time Complexity $O_S[n]$ is bounded by polynomial time complexity if

$$\exists K, C, \lambda_1 ... \lambda_K \quad :$$
$$O_S[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

## 13.2 Polynomial Space Problems

Define $\mathbb{S}^+_{space}$, the set of solutions that can be solved with polynomial time complexity

$$\mathbb{S}^+_{space} := \{s_1, s_2, ...\} :$$
$$\exists K, C, \lambda_1 ... \lambda_K :$$
$$O_S[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n, s_i \in \mathbb{S}^+_{time}$$

## 13.3 Total Polynomial Complexity Implies Space bounded Polynomial Complexity

## 13.4 Space Bounded Polynomial Complexity Implies Total Polynomial Complexity

## 13.5 Polynomial Space Complexity iff Polynomial Complexity

## 13.6 Property of Polynomial Space Complexity 1

$$lim_{n \to \infty} \frac{O_S[n+1]}{O_S[n]} = 1$$

### 13.6.1 Proof

## 13.7 Property of Polynomial Space Complexity 2

$$\exists K, C, \lambda_1, ..., \lambda_K :$$
$$(O_S[n+1] - O_S[n]) < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \;\; \forall n \;\; \forall n$$

### 13.7.1 Proof

## 13.8 Order of Complexity

Space complexity $O_S[n]$ is said to be on the order of $K_{max}$

$$O_S[n] < (\lambda_{K_{max}} n)^{K_{max}} + (\lambda_{K_{max}-1} n)^{K_{max}-1} ... + \lambda_1 n + C$$
$$O_S[n] \sim K_{max}$$

## 13.9 Proof of the existence of $O_{S_{opt}}$

# 14 Inductive Functions

## 14.1 Inductive Function $f_{n+1}$

$$O[n] := O_T[n] + O_S[n]$$
$$O[n+1] = O_T[n+1] + O_S[n+1]$$

$$f_{n+1}[n] := f[n] :$$
$$O[n+1] = f[n] + O[n] \quad \forall n$$

### 14.1.1 Proof of existence

Algebraic Proof

## 14.2 Inductive Space and Time Formulas

$$f^T_{n+1} := O_T[n+1] - O_T[n]$$
$$O_T[n+1] = O_T[n] + f^T_{n+1}$$
$$f^S_{n+1} := O_S[n+1] - O_S[n]$$
$$O_S[n+1] = O_S[n] + f^S_{n+1}$$

### 14.2.1 Proof of existence

Algebraic Proof

## 14.3 Inductive Function Expressions

Relate $f_{n+1}[n]$ to equivalence functions

$$D \in \mathbb{P}$$
$$O[n] := O_T[n] + O_S[n]$$
$$O[n+1] = O_T[n+1] + O_S[n+1] = O[n] + f_{n+1}[n]$$
$$O_T[n] = O[n] - O_S[n]$$
$$O_S[n] = O[n] - O_T[n]$$

$$f_{n+1} = O[n+1] - O[n]$$
$$f_{n+1} = O_T[n+1] + O_S[n+1] - O[n]$$

$$f_{n+1} = O_T[n+1] - O_T[n] + O_S[n+1] - O_S[n]$$
$$f_{n+1} = O[n+1] - O_T[n] - O_S[n]$$
$$f_{n+1}[n] = f^T_{n+1}[n] + f^S_{n+1}[n]$$

## 14.4 Zero Order Inductive Function

$$Let \;\; O_S[n] \sim n^0$$
$$f_{n+1} = O_T[n+1] - O_T[n] + O_S[n+1] - O_S[n] = O_T[n+1] - O_T[n]$$

## 14.5 Property of Polynomial Complexity

$f_{n+1}[n]$ has order less than O[n]
$f_{n+1}[n]$ is bound by $K_{max}$ - 1

### 14.5.1 Proof

Proof by contradiction; limit doesn't converge

# 15 Duality of $O_T[n]$, $O_S[n]$?

#### 15.0.1 $O_T$ to $O_S$

Define equivalence function $f_{T \to S}$; a function converting logical operations into memory elements

$$
\begin{gathered}
f_{T \to S} := f : \\
O_S[n] = f[n, O_T[n]] \quad \forall n, s^+ \in S^+
\end{gathered}
$$

#### 15.0.2 $O_S$ to $O_T$

Define equivalence function $f_{S \to T}$; a function converting memeory elements into logical operations

$$
\begin{gathered}
f_{S \to T} := f : \\
O_T[n] = f[n, O_S[n]] \quad \forall n, s^+ \in S^+
\end{gathered}
$$

#### 15.0.3 Invertibility?

#### 15.0.4 Polynomial Bounded?

## 15.1 Efficiency Function?

Function relating the decrease in O[n] as $O_S[n]$ increases in order
$f_{S \to T}[n, O_S[n]]$ as a function of space complexity order K?

# 16 Theorem of Computational Duality?

For all Problems in P there exists a duality function
Formally define dynamic programming, Optimal polynomial complexity minimizes the difference between time and space complexity order

$$
\begin{gathered}
D \in \mathbb{P} \\
O[n] := O_T[n] + O_S[n] \\
limit_{n \to \infty} \tfrac{O[n+1]}{O[n]} = 1 \quad \forall s^+ \in S^+_{\mathbb{P}}
\end{gathered}
$$

$$
\begin{gathered}
O_T[n] = f_{S \to T}[n, O_S[n]] \\
O_S[n] = f_{T \to S}[n, O_T[n]]
\end{gathered}
$$

$$limit_{n\to\infty}\frac{O_T[n+1]+O_S[n+1]}{O_T[n]+O_S[n]} = 1 \quad \forall s^+ \in S_{\mathbb{P}}^+$$

$$limit_{n\to\infty}\frac{f_{S\to T}[n+1,O_S[n+1]]+O_S[n+1]}{f_{S\to T}[n,O_S[n]]+O_S[n]} = 1 \quad \forall s^+ \in S_{\mathbb{P}}^+$$

$$limit_{n\to\infty}\frac{O_T[n+1]+f_{T\to S}[n+1,O_T[n+1]]}{O_T[n]+f_{T\to S}[n,O_T[n]]} = 1 \quad \forall s^+ \in S_{\mathbb{P}}^+$$

## 17 Divergent Problems

### 17.1 Definition

$$\hat{\mathcal{D}} := \{\hat{D}_1, \hat{D}_2, ...\} :$$
$$lim_{n\to\infty}\frac{O[n+1]}{O[n]} \ \ diverges \ \ \forall \hat{D} \in \hat{\mathcal{D}}$$

### 17.2 Theorem of Divergent Subfunctions

If an (inductive) subfunction of $s^+$ diverges, the solution is divergent

$$f_{n+1} = \sum g_{n+1}$$
$$limit\frac{g[n+1]}{g[n]}diverges$$

$$\exists g_{n+1} : limit\frac{g_{n+1}}{O[n]}diverges \Longrightarrow limit\frac{O[n+1]}{O[n]}diverges$$

### 17.3 Proof

### 17.4 The Set of Polynomial Solutions and the Set of Divergent Solutions are disjoint

$$\mathbb{P} \cap \hat{D} = \varnothing$$

### 17.5 Proof

Proof by contradiction; Let $s^+ \in \mathbb{P}, \hat{D}; s^+ \in \mathbb{P} \cap \hat{D}$

$$X_i = \{x_1, ..., x_n\}$$
$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

$$Let \ \ D \in \mathbb{P}$$
$$s^+ := P[X_i] \to y_o : y_o = a_o \quad \forall X_i$$
$$O[n] = O_T[n] + O_S[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} ... + \lambda_1 n + C \quad \forall n$$

# 18 Subfunctions

## 18.1 Restate the subfunction condition of general solutions

Recall the definition of general solution $s^+$

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$
$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \;\; \forall X_i) \;\; \cap \;\; (P[\hat{X}_i] \supseteq P[X_i] \;\; \forall X_i, \hat{X}_i)$$

The subfunction condition is one of two conditions for a general solution

$$P[\hat{X}_i] \supseteq P[X_i] \;\; \forall X_i, \hat{X}_i$$

## 18.2 Prove O[n] is a non-decreasing function

Consider solution $s^+$ with complexity $O[n]$

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$
$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \;\; \forall X_i) \;\; \cap \;\; (P[\hat{X}_i] \supseteq P[X_i] \;\; \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$O[n] := O_T[n] + O_S[n]$$
$$O_T[n] := |\mathcal{L}| = N$$
$$O_S[n] := |\mathcal{M}| + |y_o| = M + 1$$

O[n+1] denotes the total complexity for solution $s^+[\hat{X}_i]$

$$s^+[\hat{X}_i] = \hat{s}^+$$

Let

$$O[n+1] < O[n]$$
$$\Rightarrow \hat{N} + \hat{M} < N + M$$
$$\hat{s}^+ = \{s_1, s_2, ..., s_{\hat{N}} | b_1, b_2, ..., b_{\hat{M}}, y_o\}$$

$$\Rightarrow \hat{s}^+ \not\supseteq s^+$$
$$P[\hat{X}_i] \not\supseteq P[X_i] \quad \forall X_i, \hat{X}_i$$

$$\therefore O[n+1] < O[n] \text{ contradicts the definition of solution } s^+$$
$$O[n+1] \geqslant O[n]$$

## 18.3 Definition of Subfunction

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$
$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \;\; \cap \;\; (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub[X_i] := S = \{s_j, ... | b_k, ..., y_o\} :$$
$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S$$

### 18.3.1 $s^+[X_i]$ is a subfunction of $s^+[\hat{X}_i]$

$$s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$\hat{s}^+ = \{s_1, s_2, ..., s_{\hat{N}} | b_1, b_2, ..., b_{\hat{M}}, y_o\}; \;\; \hat{N} + \hat{M} \geqslant N + M$$

By definition of solution

$$\hat{s}^+ = P[\hat{X}_i] \supseteq P[X_i] = s^+ \quad \forall X_i, \hat{X}_i$$
$$\Rightarrow s_j, b_k \in \hat{s}^+ \;\; \forall s_j, b_k \in s^+$$

## 18.4 Subfunction Decomposition Theorem

All solutions $s^+$ can be written as the union of subfunctions $Sub_k[X_i]$

$$O[n] = O_{T_1}[n] + O_{T_2}[n] + ... + O_{T_z}[n] + |O_{S_1}[n] \cup O_{S_2}[n] \cup ... \cup O_{S_z}[n]|$$
$$O[n] = \sum |\mathcal{L}| + \sum \cup \mathcal{M}$$

## 18.5 Theorem of Divergent Subfunctions

### 18.5.1 $limit_{n\to\infty} \frac{O[n+1]}{O[n]}$ diverges $\Rightarrow$ $\exists Sub_h[X_i] : limit_{n\to\infty} \frac{O_h[n+1]}{O_h[n]}$ diverges

If any subfunction of $s^+$ diverges, then O[n+1]/O[n] diverges, $f_{n+1}$/O[n] diverges Consider solution $s^+$ with polynomial total complexity O[n] containing $z$ subfunctions $Sub_k[X_i]$ k = 1..z
FIX!!! concerns about OS memory complexity

$$X_i = \{x_1, ..., x_n, C\}; \;\; \hat{X}_i = \{x_1, ..., x_{n+1}, C\}$$
$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \;\; \forall X_i) \;\; \cap \;\; (P[\hat{X}_i] \supseteq P[X_i] \;\; \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub_h[X_i] := S_h = \{s_j, ... | b_k, ..., y_o\} :$$
$$s_j, b_k \in s^+ \;\; \forall s_j, b_k \in S_h$$

$$s^+ = Sub_1[X_i] \cup Sub_2[X_i] \cup ... \cup Sub_z[X_i]$$

$$O[n] = O_{T_1}[n] + O_{T_2}[n] + ... + O_{T_z}[n] + |O_{S_1}[n] \cup O_{S_2}[n] \cup ... \cup O_{S_z}[n]|$$
$$= O_{T_1}[n] + O_{T_2}[n] + ... + O_{T_z}[n] + O_S[n]$$

By defintion of divergent complexity

$$limit_{n\to\infty} \frac{O[n+1]}{O[n]} \text{ diverges}$$

Suppose there does not exist a diverging subfunction $Sub_h[X_i]$ for all h

$$\nexists Sub_h[X_i] :$$

$$limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]}\text{ diverges }\quad\forall h$$
$$\Rightarrow limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]}=c_h\ \ \forall h$$
$$limit_{n\to\infty}\frac{O_1[n+1]+O_2[n+1]+...+O_z[n+1]}{O_1[n]+O_2[n]+...+O_z[n]}$$

Let

$$g_h[n]=\sum_{i\neq h}O_i[n]\geqslant 0^*$$
$$\Rightarrow 0\leqslant limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]+g_h[n]}\leqslant c_h$$
$$limit_{n\to\infty}\frac{O_1[n+1]}{O_1[n]+g_1[n]}+\frac{O_2[n+1]}{O_2[n]+g_2[n]}+...+\frac{O_z[n+1]}{O_1[n]+g_z[n]}$$
$$0\leqslant limit_{n\to\infty}\frac{O_1[n+1]}{O_1[n]+g_1[n]}+\frac{O_2[n+1]}{O_2[n]+g_2[n]}+...+\frac{O_z[n+1]}{O_1[n]+g_z[n]}\leqslant\sum_{i=1}^{z}c_i$$
$$\Rightarrow limit_{n\to\infty}\frac{O_1[n+1]}{O_1[n]+g_1[n]}+\frac{O_2[n+1]}{O_2[n]+g_2[n]}+...+\frac{O_z[n+1]}{O_1[n]+g_z[n]}=\tilde{C}$$
$$0\leqslant\tilde{C}\leqslant\sum_{i=1}^{z}c_i$$

*O$_i[n]\geqslant 0$ is a non-decreasing function

Assuming

$$\nexists Sub_h[X_i]:$$
$$limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]}\text{ diverges }\quad\forall h$$
$$\Rightarrow limit_{n\to\infty}\frac{O_[n+1]}{O_1[n]}=\tilde{C}$$

Contradicting the definition of divergent solution

$$\therefore\exists Sub_h[X_i]:$$
$$limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]}\text{ diverges}$$

### 18.5.2 $\exists Sub_h[X_i]: limit_{n\to\infty}\frac{O_h[n+1]}{O_h[n]}$ diverges $\Rightarrow$ $limit_{n\to\infty}\frac{O[n+1]}{O[n]}$ diverges

$$X_i=\{x_1,...,x_n,C\};\ \ \hat{X}_i=\{x_1,...,x_{n+1},C\}$$

$$s^+ = s^+[X_i] := P :$$
$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, ..., s_N | b_1, b_2, ..., b_M, y_o\} = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\}$$
$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub_h[X_i] := S_h = \{s_j, ... | b_k, ..., y_o\} :$$
$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S_h$$

$$s^+ = Sub_1[X_i] \cup Sub_2[X_i] \cup ... \cup Sub_z[X_i]$$
$$O[n] = O_{T_1}[n] + O_{T_2}[n] + ... + O_{T_z}[n] + |O_{S_1}[n] \cup O_{S_2}[n] \cup ... \cup O_{S_z}[n]|$$
$$= O_{T_1}[n] + O_{T_2}[n] + ... + O_{T_z}[n] + O_S[n]$$

Suppose

$$\exists Sub_h[X_i] : limit_{n \rightarrow \infty} \tfrac{O_h[n+1]}{O_h[n]} \text{ diverges}$$

$$limit_{n \rightarrow \infty} \tfrac{O[n+1]}{O[n]}$$
$$= limit_{n \rightarrow \infty} \tfrac{O_1[n+1] + O_2[n+1] + ... + O_z[n+1]}{O_1[n] + O_2[n] + ... + O_z[n]}$$
$$limit_{n \rightarrow \infty} \tfrac{O_1[n+1]}{O[n]} + ... + \tfrac{O_h[n+1]}{O[n]} + ... + \tfrac{O_z[n+1]}{O[n]}$$

## 19 Sum to N Problem with 2 integers

### 19.1 State formal definition of Sum to N : $x_i + x_j == N$

$$X_i = \{x_1, ..., x_n, N\}$$
$$D := f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$
$$s^+ = P :$$
$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$
$$s^+ = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$D = f[X_i] = \exists x_j, x_k \in X_i : x_j + x_k == N$$

### 19.2 Express a formal solution : $O_S[n] \sim n^0$

$$s^+ = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$
$$s_1 = y_o \leftarrow \mathbb{F};$$
$$\forall i, j > i$$
$$s_2, s_3, s_8, s_9, ..., s_{3ij-4}, s_{3ij-3} ..., s_{3n(n-1)-4}, s_{3n(n-1)-3} = b_1 \leftarrow x_i + x_j$$
$$s_4, s_5, s_{10}, s_{11}, ..., s_{3ij-2}, s_{3ij-1} ..., s_{3n(n-1)-2}, s_{3n(n-1)-1} = b_1 \leftarrow b_1 == N$$
$$s_6, s_7, s_{12}, s_{13} ..., s_{3ij}, s_{3ij+1} ..., s_{3n(n-1)}, s_{3n(n-1)+1} = y_o \leftarrow y_o \vee b_1$$
$$s^+ = \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \quad \forall i, j > i \mid b_1, y_o\}$$

### 19.3 Show $s^+$ satisfies the subfunction condition of solutions: $P[\hat{X}_i] \supseteq P[X_i] \quad \forall \hat{X}_i, X_i$

$$X_i = \{x_1, ..., x_n, N\}; \quad \hat{X}_i = \{x_1, ..., x_n, x_{n+1}, N\}$$
$$s^+ = \{s_1, s_2, ..., s_{O_T[n]}, b_1, b_2, ..., b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$
$$s^+_{n+1} = s^+ \cup \hat{s}^+$$
$$s_1 = y_o \leftarrow \mathbb{F};$$
$$\forall i, j > i$$
$$s_2, s_3, s_8, s_9, ..., s_{3ij-4}, s_{3ij-3} ..., s_{3n(n-1)-4}, s_{3n(n-1)-3} = b_1 \leftarrow x_i + x_j$$
$$s_4, s_5, s_{10}, s_{11}, ..., s_{3ij-2}, s_{3ij-1} ..., s_{3n(n-1)-2}, s_{3n(n-1)-1} = b_1 \leftarrow b_1 == N$$

$$s_6, s_7, s_{12}, s_{13}..., s_{3ij}, s_{3ij+1}..., s_{3n(n-1)}, s_{3n(n-1)+1} = y_o \leftarrow y_o \vee b_1$$

$$\forall k$$

$$s... = b_1 \leftarrow x_k + x_{n+1}$$

$$s... = b_1 \leftarrow b_1 == N$$

$$s... = y_o \leftarrow y_o \vee b_1$$

$$s^+ = \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \quad \forall i, j > i \mid b_1, y_o\}$$

$$\hat{s}^+ = \{y_o \leftarrow y_o \vee (x_k + x_{n+1} == N) \ \forall k < n+1 | \ b_1, y_o\}$$

$$s^+_{n+1} = \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \quad \forall i, j > i \mid b_1, y_o\} \ \cup$$

$$\{y_o \leftarrow y_o \vee (x_k + x_{n+1} == N) \ \forall k < n+1 | \ b_1, y_o\}$$

$$s^+_{n+1} = P[\hat{X}_i] \supseteq P[X_i] = s^+$$

### 19.4 Determine $O[n], O_S[n], O_T[n], \hat{O}[n], \hat{O}_T[n], \hat{O}_S[n]$ for the above solution

$$O_S[n] = |y_o| + |b_1| = 2$$

$$O_T[n] = 3n(n-1) + 1 = 3n(n-1) - 1 + O_S[n]$$

$$O[n] = 3n(n-1) + 3 = 3n^2 - 3n + 3$$

$$\hat{O}_S[n] = 0$$

$$\hat{O}_T[n] = 6n$$

$$\hat{O}[n] = \hat{O}_S[n] + \hat{O}_T[n]$$

### 19.5 Verify $O[n+1] = O[n] + \hat{O}[n]$

$$O[n+1] = O[n] + \hat{O}[n]$$

$$3(n+1)^2 - 3(n+1) + 3 = 3n^2 - 3n + 3 + 6n$$

$$3n^2 + 6n + 3 - 3n - 3 + 3 = 3n^2 + 3n + 3$$

$$3n^2 + 3n + 3 = 3n^2 + 3n + 3$$

## 19.6 Show $s^+$ has Polynomial Complexity by the definition of Total Polynomial Complexity

$$O[n] = 3n^2 - 3n + 3$$

## 19.7 Show $s^+$ has Polynomial Complexity by showing limit$_{n\to\infty}\frac{O[n+1]}{O[n]}$ = 1

$$limit_{n\to\infty}\frac{O[n+1]}{O[n]} =$$

$$limit_{n\to\infty}\frac{3n^2+3n+3}{3n^2-3n+3} =$$

$$limit_{n\to\infty}\left(\frac{3n^2-3n+3}{3n^2-3n+3} + \frac{6n}{3n^2-3n+3}\right) =$$

$$limit_{n\to\infty}\left(1 + \frac{6n}{3n^2-3n+3}\right) = 1$$

## 20 Divergent Problems

### 20.1 Definition

$$\hat{\mathcal{D}} := \{\hat{D}_1, \hat{D}_2, ...\} :$$
$$lim_{n\to\infty}\frac{O[n+1]}{O[n]}\ \ diverges\ \ \forall \hat{D} \in \hat{\mathcal{D}}$$

### 20.2 Theorem of Divergent Subfunctions

If an (inductive) subfunction of $s^+$ diverges, the solution is divergent

$$f_{n+1} = \sum g_{n+1}$$
$$limit\frac{g[n+1]}{g[n]}diverges$$

$$\exists g_{n+1} : limit\frac{g_{n+1}}{O[n]}diverges \Longrightarrow limit\frac{O[n+1]}{O[n]}diverges$$

### 20.3 Proof

### 20.4 The Set of Polynomial Solutions and the Set of Divergent Solutions are disjoint

$$\mathbb{P} \cap \hat{D} = \varnothing$$

### 20.5 Proof

Proof by contradiction; Let $s^+ \in \mathbb{P}, \hat{D}; s^+ \in \mathbb{P} \cap \hat{D}$

$$X_i = \{x_1, ..., x_n\}$$
$$D := f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\}\quad \forall X_i$$

$$Let\ \ D \in \mathbb{P}$$

$$s^+ := P[X_i] \rightarrow y_o : y_o = a_o\quad \forall X_i$$
$$O[n] = O_T[n] + O_S[n] < (\lambda_K n)^K + (\lambda_{K-1}n)^{K-1}... + \lambda_1 n + C\quad \forall n$$

# Traveling Salesman Problem of Dimension 2

## 21 Proof of the existence of $\hat{\mathcal{D}}$

### 21.1 The Traveling Salesman Problem of Dimension 2

English description

### 21.2 Formal Definition

$$X_i = \{l_1, l_2, ..., l_n, C\}$$

$$l_i = \{x_i, y_i\} \ \ \forall i$$

$l_i$ denotes the 2D coordinates of location i

$$C = \{d_{proposed}, p_{decimal}\}$$

$d_{proposed}$ denotes the suggested shortest distance

$p_{decimal}$ is the decimal precision

$$L[l_i, l_j] := \sqrt{(y_j - y_i)^2 + (x_j - x_i)^2}$$

Let $L[l_i, l_j]$ denote the distance between location $l_i$ and $l_j$

$$\tilde{L}[l_i, l_j] := d_{trunc} : -p_{decimal} < d_{trunc} - L[l_i, l_j] < p_{decimal}$$

Let $\tilde{L}[l_i, l_j]$ denote a truncated decimal representation of $L[l_i, l_j]$

$$R_i := \{r_1, r_2, ..., r_n, r_1\} \ \ : \ \ r_i \in X_i \ \ \forall i; \ \ r_i \neq r_j$$

Let $R_i$ denote route $i$

$$L_{Total}[R_i] := (\textstyle\sum_{i=1}^{n-1} \tilde{L}[r_i, r_{i+1}]) + \tilde{L}[r_n, r_1]$$

Let $L_{Total}[R_i]$ denote the sum of truncated lengths of route $R_i$

$$D := f[X_i] \to a_o \in \{\mathbb{T}, \mathbb{F}\} \ \ \forall X_i$$

$$a_o =$$

$$(\exists R_k : L_{total}[R_k] == d_{proposed}) \cap (\nexists R_j : L_{total}[R_j] < d_{proposed})$$

## Traveling Salesman Problem of Dimension 2

### 21.3 Define subpath, subpath distance, subpath storage

$\tilde{L}[l_i, l_j]$ denotes "the distance of a subpath of length 1"

$$\tilde{L}[l_i, l_j] := d_{trunc} : -p_{decimal} < d_{trunc} - L[l_i, l_j] < p_{decimal}$$
$$= abs(d_{trunc} - L[l_i, l_j]) < p_{decimal}$$

$\tilde{R}$ denotes a subpath of length k

$$\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, ..., \tilde{r}_k\} \;\; : \;\; \tilde{r}_i \in X_i \;\; \forall i, r_i \neq r_j$$

$\tilde{L}_k[\tilde{R}]$ denotes "the distance of a subpath of length k"

$$\tilde{L}_k[\tilde{R}] := \textstyle\sum_{i=1}^{k} \tilde{L}[\tilde{r}, \tilde{r}_{i+1}]$$

Let $\mathcal{M}_1$ denote the memory reserved for subpaths distances of length 1

$$\mathcal{M}_1 = \{\hat{b}_{1;1}, \hat{b}_{1;2}, \hat{b}_{1;3}, ..., \hat{b}_{startindex;finishindex}, ..., \hat{b}_{n-1;n}\}^*$$
$$\mathcal{M} \supseteq \mathcal{M}_1$$

* Note $\hat{b}_{i;j} = \hat{b}_{j;i}$
$\sqrt{(y_j - y_i)^2 + (x_j - x_i)^2} = \sqrt{(y_i - y_j)^2 + (x_i - x_j)^2}$

### 21.4 Define the following functions

#### 21.4.1 $sqrt[x, p_{decimal}] = \sqrt{x}$ **[1]**

#### 21.4.2 $pow[x, 2, p_{decimal}] = x^2$ **[2]**

### 21.5 Define the following subfunctions

#### 21.5.1 loadM1Subpaths$[X]$

// Computes all subpaths of length 1 and stores in $\mathcal{M}_1 = \{\hat{b}_{1;1}, \hat{b}_{1,;2}, ..., \hat{b}_{n-1;n}\}$

$$//X_i = \{l_1, l_2, ..., l_n, C\}$$
$$//l_i = \{x_i, y_i\} \;\; \forall i$$

$$//\mathcal{M} = \{b_1, b_2, ..., b_M, \hat{b}_{1;1}, \hat{b}_{1,;2}, ..., \hat{b}_{n-1;n}, y_o\} = \{b_1, b_2, ..., b_M, \mathcal{M}_1, y_o\} = \{\mathcal{M}, \mathcal{M}_1, y_o\}$$

$$\forall i, j > i$$

$$b_3 \leftarrow y_i - y_j$$

$$b_4 \leftarrow x_i - x_j$$

$$b_3 \leftarrow b_3^2$$

$$b_4 \leftarrow b_4^2$$

$$b_3 \leftarrow b_3 + b_4$$

$$\hat{b}_{i;j} \leftarrow \sqrt{b_3}^*$$

${}^*\hat{b}_{i;j} = \tilde{L}[l_i, l_j]$

### 21.5.2 computeAllRoutes$[X]$

// Computes all complete routes, checks for a route $==$ $d_{proposed}$, sets $y_o$ to false if the current route is shorter than $d_{proposed}$

$$\forall i, j \neq i, k \neq i, j..., q \neq i, j, ..., m$$

$$b_3 \leftarrow \hat{b}_{1;j} + \hat{b}_{j;k}$$

$$b_3 \leftarrow b_3 + \hat{b}_{k;l}$$

$$...$$

$$b_3 \leftarrow b_3 + \hat{b}_{m;q}$$

$$b_3 \leftarrow b_3 + \hat{b}_{q;1}$$

$$b_4 \leftarrow b_3 == b_2$$

$$b_1 \leftarrow b_1 \vee b_4$$

$$b_4 \leftarrow b_2 \leqslant b_3$$

$$y_o \leftarrow y_o \wedge b_4$$

## 21.6 Express a solution using subfunctions, storing subpaths of length 1 in memory

// $d_{proposed}$ is the shortest path

$$y_o \leftarrow \mathbb{T}$$

$$//d_{proposed} \text{ exists as a total path length}$$
$$b_1 \leftarrow \mathbb{F}$$

$$// \text{ shortest path register}$$
$$b_2 \leftarrow d_{proposed}$$

$$loadM1Subpaths[X]$$
$$computeAllRoutes[X]$$

### 21.7 Show each subfunction satisfies the subfunction condition of solutions : $P[\hat{X}_i] \supseteq P[X_i] \;\; \forall \hat{X}_i, X_i, \;\; \hat{X}_i \supseteq X_i$

Let

$$\mathcal{M}_0 = \{b_1, b_2, b_3, b_4, y_o\}$$
$$\mathcal{M}_1 = \{\hat{b}_{1;1}, \hat{b}_{1,;2}, ..., \hat{b}_{n-1;n}\}$$

#### 21.7.1 $loadM1Subpaths[X] \rightarrow \mathcal{M}_1$

Let

$$//X = \{l_1, l_2, ..., l_n, C\}; \;\; \hat{X} = \{l_1, l_2, ..., l_n, l_{n+1}, C\}$$

$$loadM1Subpaths[X, \mathcal{M}] \rightarrow \mathcal{M}_1 = Sub_1[X, \mathcal{M}] \rightarrow \mathcal{M}_1$$

$$Sub_1[X, \mathcal{M}] = \{\mathcal{L}, \mathcal{M}\}$$
$$= \{\hat{b}_{i;j} \leftarrow \tilde{L}[l_i, l_j] \;\; \forall i, j > i | b_3, b_4, \hat{b}_{1;1}, \hat{b}_{1;2}, ..., \hat{b}_{n-1;n}\}$$
$$Sub_1[X_i, \mathcal{M}] = \{\hat{b}_{i;j} \leftarrow \tilde{L}[l_i, l_j] \;\; \forall i, j > i | b_3, b_4, \mathcal{M}_1\}$$

$$Sub_1[\hat{X}, \mathcal{M}] = \{\hat{\mathcal{L}}, \hat{\mathcal{M}}\}$$
$$= \{\hat{b}_{i;j} \leftarrow \tilde{L}[l_i, l_j] \;\; \forall i, j > i | b_3, b_4, \hat{b}_{1;1}, \hat{b}_{1;2}, ..., \hat{b}_{n;n+1}\}$$
$$= \{\mathcal{L}, \hat{b}_{i;j} \leftarrow \tilde{L}[l_i, l_j] \;\; \forall i, j = n+1 | \mathcal{M}, \hat{b}_{1;n+1}, \hat{b}_{2;n+1}, ..., \hat{b}_{n;n+1}\}$$
$$Sub_1[\hat{X}, \mathcal{M}] = \{\mathcal{L}, \mathcal{L}_{n+1} | \mathcal{M}, \mathcal{M}_{n+1}\}$$
$$Sub_1[\hat{X}, \mathcal{M}] = \{\mathcal{L}, \mathcal{L}_{n+1} | \mathcal{M}, \mathcal{M}_{n+1}\} \supseteq \{\mathcal{L} | \mathcal{M}\} = Sub_1[X, \mathcal{M}]$$

### 21.7.2 $computeAllRoutes[X]$

Let

$$computeAllRoutes[X] = Sub_2[X]$$

$$Sub_2[X] = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$= \{\hat{b}_{1;i_2} + \hat{b}_{i_2;i_3} + \hat{b}_{i_3;i_4} + ... + \hat{b}_{i_n;1} \ \ \forall i_2, i_3 \neq i_2, i_4 \neq i_2, i_3 ... i_n \neq i_2, i_3 ..., i_{n-1}$$
$$|b_1, b_2, b_3, b_4, \mathcal{M}_1, y_o\}$$

$$Sub_2[\hat{X}] = \{\hat{\mathcal{L}}, \hat{\mathcal{M}}, y_o\}$$

$$= \{\hat{b}_{1;i_2} + \hat{b}_{i_2;i_3} + \hat{b}_{i_3;i_4} + ... + \hat{b}_{i_{n+1};1} \ \ \forall i_2, i_3 \neq i_2, i_4 \neq i_2, i_3 ... i_{n+1} \neq i_2, i_3 ..., i_n$$
$$|\mathcal{M}, \mathcal{M}_{n+1}, y_o\}$$

Let

$$insert\_subpath[\mathcal{L}] =$$

$$Sub_2[\hat{X}] = \{insert\_subpath[\mathcal{L}, \hat{b}_{i_{n+1};j}, j] \ \ \forall j \neq n+1 | \mathcal{M}, \mathcal{M}_{n+1}, y_o\}$$

### 21.7.3 Show the overall solution storing subpaths of length 1 satisfies the subfunction condition of solutions : $P[\hat{X}_i] \supseteq P[X_i] \ \ \forall \hat{X}_i, X_i$

## 21.8 Express $O[n]$ in terms of subfunction complexities

$$O_{sub1}[n] = O_{T_{sub1}}[n] + O_{S_{sub1}}[n]$$
$$O_{sub2}[n] = O_{T_{sub2}}[n] + O_{S_{sub2}}[n]$$
$$O[n] = O_{sub1}[n] + O_{sub2}[n] + 3$$

## 21.9 $\mathbf{Sub_+[X]}$

Let $Sub_+[X]$ denote a subfunction that adds all subpaths of length 1
Let $O_+[n]$ denote the total complexity of subfunction $Sub_+[X]$

### 21.9.1 Find an expression for $O_+[n]$ := the number of $\tilde{L}[l_i, l_j] + \tilde{L}[l_j, l_k]$ length 1 subpath additions

$$O_+[n] = (\textstyle\sum_{i=1}^{n} 1)\frac{(nP(n-1)}{2}$$

$$O_+[n] = \frac{n(n-1)!}{2}$$
$$O_+[n] = \frac{n!}{2}$$

## 21.10 Prove $Sub_+[X]$ is a subfunction of all $s^+$ by contradiction

suppose not all subpaths are considered
there could exist subpath resulting in an incorrect solution
contradicts definition of solution

## 21.11 Show the solution storing subpaths of length 1 contains Sub$_+$[X]

$$// \ d_{proposed} \text{ is the shortest path}$$
$$y_o \leftarrow \mathbb{T}$$

$$//d_{proposed} \text{ exists as a total path length}$$
$$b_1 \leftarrow \mathbb{F}$$

$$// \text{ shortest path register}$$
$$b_2 \leftarrow d_{proposed}$$

$$loadM1Subpaths[X]$$
$$computeAllRoutes[X]$$

## 21.12 Express $O[n]$ in terms of subfunction complexities including $O_+[n]$ as a subfunction complexity

$$O[n] = O_{sub1}[n] + O_{sub2}[n] + 3$$
$$O_{sub2}[n] = O_+[n] + 8\frac{nP(n-1)}{2}$$
$$O_{sub2}[n] = O_+[n] + 8\frac{(n-1)!}{2}$$
$$O[n] = O_{sub1}[n] + O_+[n] + 8\frac{(n-1)!}{2} + 3$$

## 21.13 Show $limit_{n\to\infty}\frac{O_+[n+1]}{O_+[n]}$ diverges

$$limit_{n\to\infty}\frac{O_+[n+1]}{O_+[n]}$$

$$= limit_{n \to \infty} \frac{(n+1)!}{2} \frac{2}{n!}$$
$$= limit_{n \to \infty} \, n$$

There does not exist ...
therefore $limit_{n \to \infty} \frac{O_+[n+1]}{O_+[n]}$ diverges

## 21.14 Prove D diverges by the theorem of divergent subfunctions

## 21.15 Connection to "P $\neq$ NP"

## Citations


[1] *chatgpt*
[2] *https://stackoverflow.com/questions/*3518973*/floating−point−exponentiation−without − power − function*
[3] *https://stackoverflow.com/questions/*27086195*/linear−index−upper−triangular − matrix*
[4] *http://www.math.uchicago.edu/ may/VIGRE/VIGRE*2011*/REUPapers/Riffer−Reinert.pdf*