פתרון תרגיל בית מספר 2

Firing Squad Problem

Firing squad synchronization problem

From Wikipedia, the free encyclopedia

The **firing squad synchronization problem** is a problem in computer science and cellular automata first proposed by John Myhill in 1957 and published (with a solution) in 1962 by Edward Moore. The problem is analogous to problems of logical design, systems design, and programming, and can be stated as follows:

[edit] History

The first solution to the FSSP was found by John McCarthy and Marvin Minsky and was published in Sequential Machines by Moore. A solution using a minimal amount of states was introduced by Jacques Mazoyer in 1988, whose solution uses only six states.[1] In addition, he also proved that no four state solution exists. It is still unknown whether a five state solution exists.

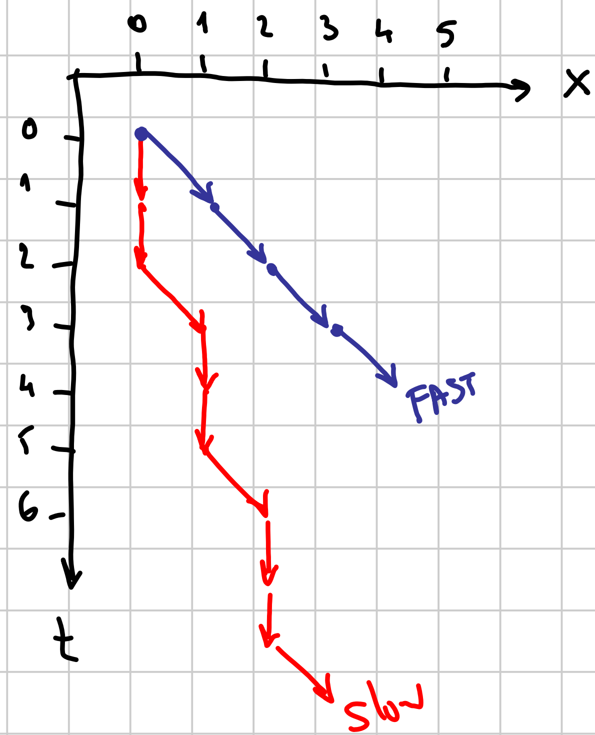
A solution using a minimal amount of time was later found by Professor E. Goto at M.I.T., whose solution uses thousands of states and requires exactly $2n - 2$ units of time for n soldiers. It is proven that a solution using a smaller amount of time cannot exist.

[edit] General solution

A general solution to the FSSP involves propagating two waves down the line of soldiers: a fast wave and a slow wave moving three times as slow. The fast wave bounces off the other end of the line and meets the slow wave in the centre. The two waves then split into four waves, a fast and slow wave moving in either direction from the centre, effectively splitting the line into two equal parts. This process continues, subdividing the line until each division is of length 1. At this moment, every soldier fires. This solution requires $3n$ units of time for n number of soldiers.

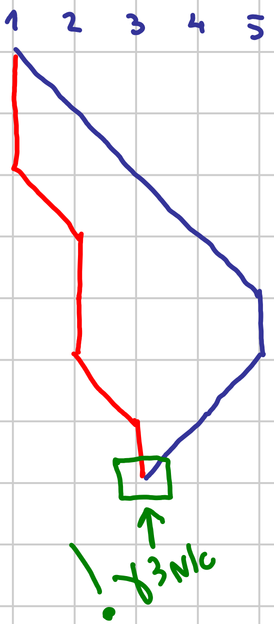
מציאת מעגל

(נניח אורך n)



מצביע $\leftrightarrow s_1 \leftrightarrow s_2 \leftrightarrow \dots \leftrightarrow s_n$

שימוש בהליכה מהירה והליכה איטית.



מפגש במעגל:

$$3 \cdot t\, v_{slow} = t \cdot v_{fast}$$
$$\Downarrow$$
$$3\, v_{slow} = v_{fast}$$

רדוקציות:

1. $\text{מקור} - s_1 \cdots - s_k - s_{k+1} - \cdots - s_{2k+1} - W$

$\Downarrow$

$W - s_1 - \cdots - s_k - M - s_{k+2} \cdots - s_{2k+1} - W$

1 2 3 4 5

$2k+1$

2. $W - S^{2k+1} - M - S^{2k+1} - M - S^{2k+1} - \cdots M - S^{2k+1} - W$

$\Downarrow$

$W - S^k - M - S^k - W - S^k - M - S^k - W - S^k - M - S^k - \cdots W - S^k - M - S^k$

פלט: מציאת אמצע

סיום: מקרי הבסיס

צריך לתכנן שבכל המקרים כולם ייכנסו ל sync בזוג.

הנחות: 1) כל 3 מתי' M שולחים START ביחד.

הוכחה

2) חייל שמאלי ביותר (שגם של מפקד) יודע מאתי, ולכן אז צריך START נוסף מהמפקד

W – S – מפקד (חייל אחד)

M – S – W (חייל שמאלי)

W – S – M (חייל פנימי)

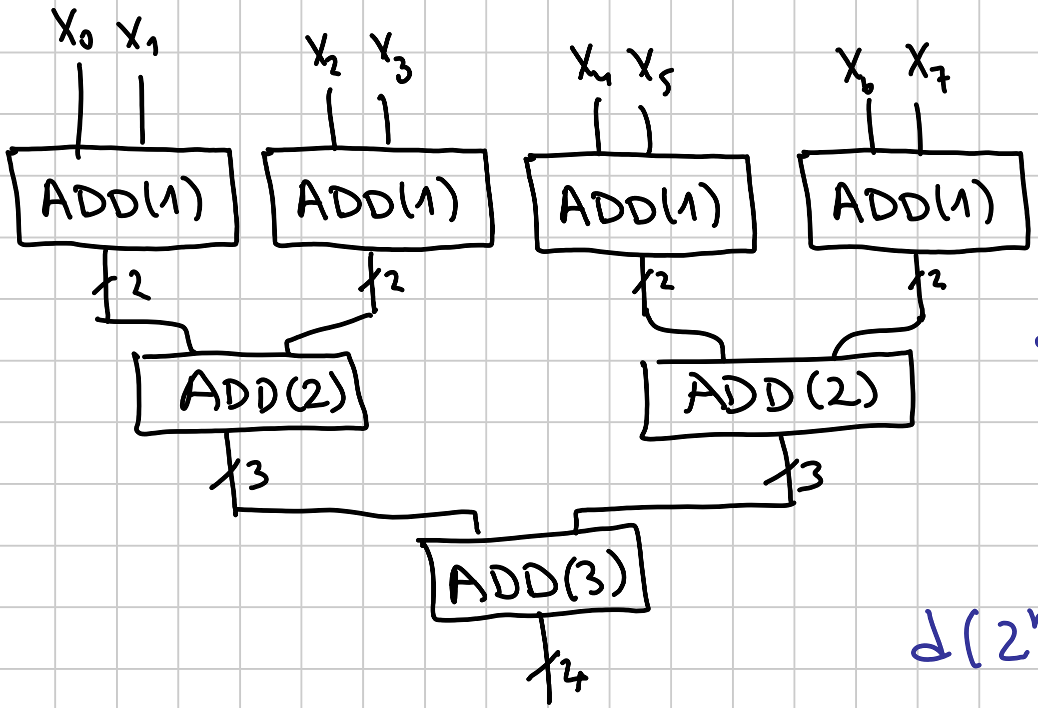
Hamming Weight

לחשב $weight(x[N-1:0])$

בזמן $O(\lg N)$

אם נבנה עץ של מחברים בינאריים

כלל נציב $N=2^n$ קלטים:



$$d(N) = d(2^n) = d(2^{n-1}) + d(ADD(n+1))$$

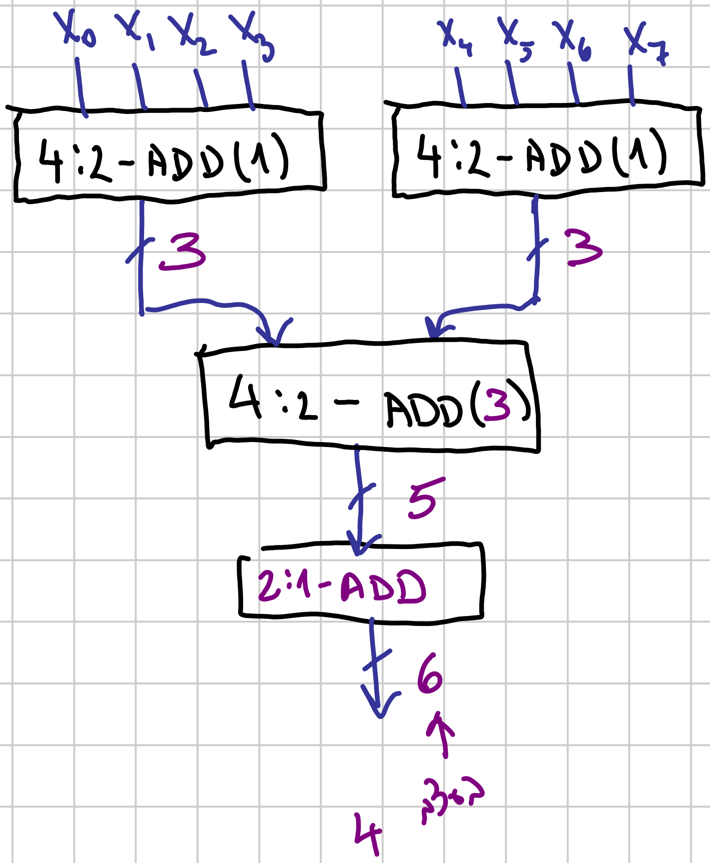
$$= d(2^{n-1}) + O(\log n)$$

$$\Leftarrow \quad d(2^n) = \Theta\left(\sum_{i=1}^{n} \log i\right)$$

$$= \Theta(n \cdot \log n)$$

$$= \Theta(\lg N \cdot \lg\lg N)$$

לכן זמן לוגריתמי!



במקום טבלת נבנה עץ של
מחברים 4:2

$$d(N)=\begin{cases} d(N/2)+\overbrace{d(4{:}2\text{-}add)}^{\text{קבוע!}} & \text{if } N>2 \\ d(ADD(2)) & \text{if } N=2 \end{cases}$$

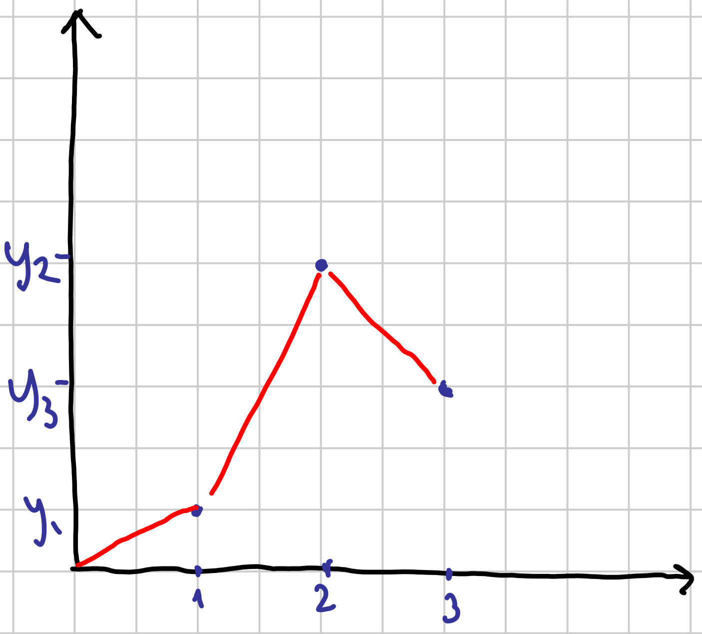
$$d(N) = O(\log N)$$

כנדרש.

בדיקת קמירות

$\{(k,y_k)\}_{k=1}^{i}$ קמור $\iff$

$$\forall k \quad \frac{y_{k+2}-y_{k+1}}{k+2 - k+1} \geq \frac{y_{k+1}-y_k}{k+1-k}$$



$$\iff y_{k+2}-y_{k+1} \geq y_{k+1}-y_k$$

נסמן $\Delta_k = y_k - y_{k-1}$. ונדרוש $\Delta_{k+1} \geq \Delta_k$, לכל $i \geq k+1$.

ולכן $convex_i = AND(\Delta_{k+1} \geq \Delta_k : k \leq i-1)$

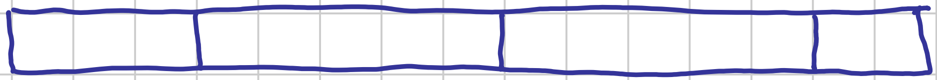
הרדוקציה: 1) חשב Δ_k לכל k.

2) חשב $\Delta_{k+1} \geq \Delta_k$ לכל $n-1 \geq k$

3) בצע $PPC-AND(n)$ כדי לחשב את

SEGMENTED PPC

נגדיר סיבית b_i המציינת האם x_i ממוקם בתחילת רצף

x 

b 1 0 0 1 0 0 0 0 1 0 0 0 0 1 0

נגדיר

$$\tilde{F} = F \times \{0,1\}$$

$$\tilde{*} : \tilde{F} \times \tilde{F} \to \tilde{F}$$

אז הכלל:

$$(f_1, b_1) \tilde{*} (f_2, b_2) = \begin{cases} (f_1 * f_2, b_1) & \text{if } b_2 = 0 \\ (f_2, 1) & \text{if } b_2 = 1 \end{cases}$$

<u>טענה:</u> $*$ אסוצ' $\Rightarrow$ $\tilde{*}$ אסוצ'

<u>הוכחה:</u> ⓐ אם $b_3 = 1$ אז

$$(f_1, b_1) \tilde{*} \left[(f_2, b_2) \tilde{*} (f_3, b_3) \right]$$

$$= (f_1, b_1) \tilde{*} (f_3, 1) = (f_3, 1)$$

$$\left[(f_1, b_1) \tilde{*} (f_2, b_2)\right] \tilde{*} (f_3, b_3)$$

$$= (f, b) \tilde{*} (f_3, 1) = (f_3, 1)$$

אז $b_3 = 0$ אם ?

$$(f_1, b_1) \tilde{*} \left[(f_2, b_2) \tilde{*} (f_3, b_3)\right]$$

$$= (f_1, b_1) \tilde{*} (f_2 * f_3, b_2)$$

$b_2 = 1$ אם

$$\left(\overset{\text{אם}}{b_2 = 1}\right) = (f_2 * f_3, 1)$$

$b_2 = 0$

$$\left(\overset{\text{אם}}{b_2 = 0}\right) = (f_1 * f_1 * f_2, b_1)$$

$$\left[(f_1, b_1) \tilde{*} (f_2, b_2)\right] \tilde{*} (f_3, b_3) =$$

אבל

אם $b_2 = 1$ $\quad = (f_2, 1) \tilde{*} (f_3, 0) = (f_2 * f_3, 1)$ $\quad$ ($b_2 = 1$)

אם $b_2 = 0$ $\quad = (f_1 * f_2, b_1) \tilde{*} (f_3, 0) = (f_1 * f_2 * f_3, b_1)$ $\quad$ ($b_2 = 0$)

<u>טענה:</u> אם $b[i:j] = 100..0$

אז $\tilde{f}_1 \tilde{*} \cdots \tilde{*} \tilde{f}_j = (f_i * \cdots * f_j, 1)$

<u>הוכחה:</u> באינדוקציה על $j - i$.

בסיס: $j = i$

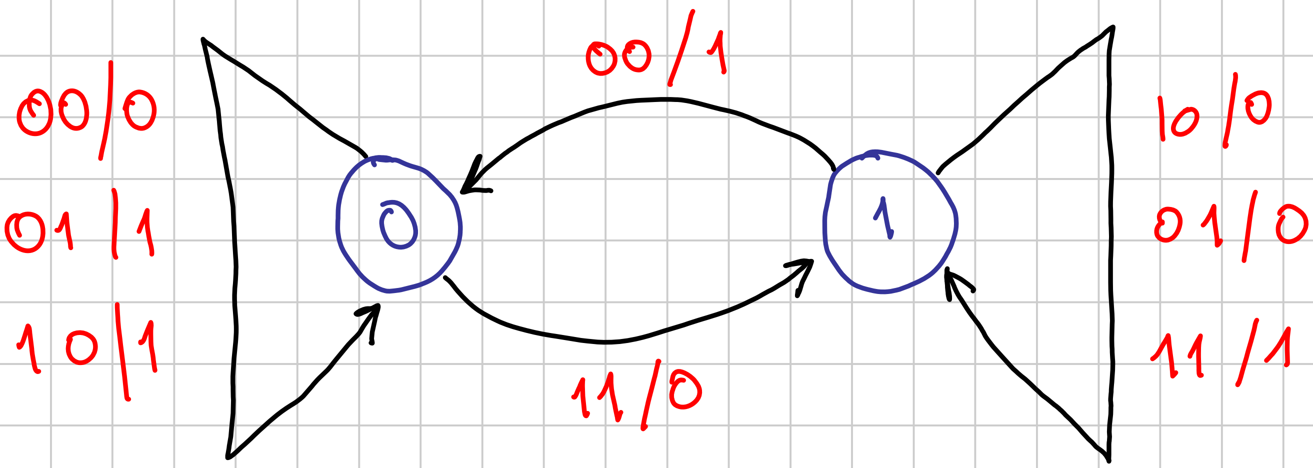
$$\tilde{f}_1 \tilde{*} \cdots \tilde{*} \tilde{f}_i = \tilde{\pi}_{i-1} \tilde{*} (f_i, 1)$$
$$= (f_i, 1)$$

שלב האינדוקציה: $j > i$

$$\tilde{f}_1 \tilde{*} \cdots * \tilde{f}_j = (\tilde{f}_1 \tilde{*} \cdots \tilde{*} \tilde{f}_{i-1} \tilde{*} \tilde{f}_i) \tilde{*} (\tilde{f}_{i+1} \tilde{*} \cdots \tilde{*} \tilde{f}_j)$$
$$= \tilde{f}_i \tilde{*} \cdots \tilde{*} \tilde{f}_j = (f_i * \cdots f_j, 1)$$

חישוב $A+B$, $A+B+1$

תזכורת: חיבור סדרתי



סדרת קלט $\{\sigma_i\}_{i=0}^{n-1}$ מגדירה מסלול בגרף המצבים

$$q_0 \to q_1 \to q_2 \to \cdots \to q_n$$

נשים לב:

$$q_1 = \delta(q_0, \sigma_0) = \delta_0(q_0)$$

$$q_2 = \delta(q_1, \sigma_1) = \delta_1(q_1) = \delta_1(\delta_0(q_0))$$

ובאופן כללי

$$q_i = \delta_{i-1}(\delta_{i-2}(\cdots \delta_0(q_0)))$$

π_i היא הרכבה של פונקציות המעבר

$$\pi_i \equiv \delta_{\sigma_i} \circ \delta_{\sigma_{i-1}} \circ \dots \circ \delta_{\sigma_0}$$

ולכן אם q_0 הוא המצב ההתחלתי אז $\pi_i(q_0) = q_{i+1}$

וכמובן $S_i = \lambda(q_i, \sigma_i)$
$= \pi_{i-1}(q_0) \oplus \sigma_i$

המצב ההתח' בחישוב $A+B$ הוא $q_0 = 0$

בשלב בחישוב $A+B+1$, המצב ההתח' הוא $q_0 = 1$

נשים לב ש $\pi_{i-1} \in \{f_0, f_1, f_{id}\}$

ולכן $(A+B+1)[i] = 0 \iff \pi_{i-1}(1) \oplus \sigma_i = 0$

בנוסף שבחישוב $A+B$: $(A+B)[i] = 0 \iff \pi_{i-1}(0) \oplus \sigma_i = 0$