טרנספורם פוריה

טרנספורם פוריה מהיר: Fast Fourier Transform (FFT)

אלגוריתם רקורסיבי מהיר, עם זמן הרצה שתלוי כך במספר הנקודות N:

$$O(N\log_2 N)$$

הרקורסיביות מחלקת כל פעם את הנקודות לשתי קבוצות שוות, לכן N חייב להיות חזקה של 2.

void four1(float data[], unsigned long nn, int isign)
Replaces data[1..2*nn] by its discrete Fourier transform, if isign is input as 1; or replaces data[1..2*nn] by nn times its inverse discrete Fourier transform, if isign is input as -1. data is a complex array of length nn or, equivalently, a real array of length 2*nn. nn MUST be an integer power of 2 (this is not checked for!).

N=nn. הווקטור data קולט את h ופולט את H (שניהם קומפלקסים!)

כדי לחשב טרנספורם: isign=1 .

כדי לחשב טרנספורם הפוך: isign=-1 , וצריכים אנחנו לחלק ב- N.

טרנספורם פוריה

כיצד שומרים N מספרים קומפלקסים (H_n או h_k) בתוך data ?

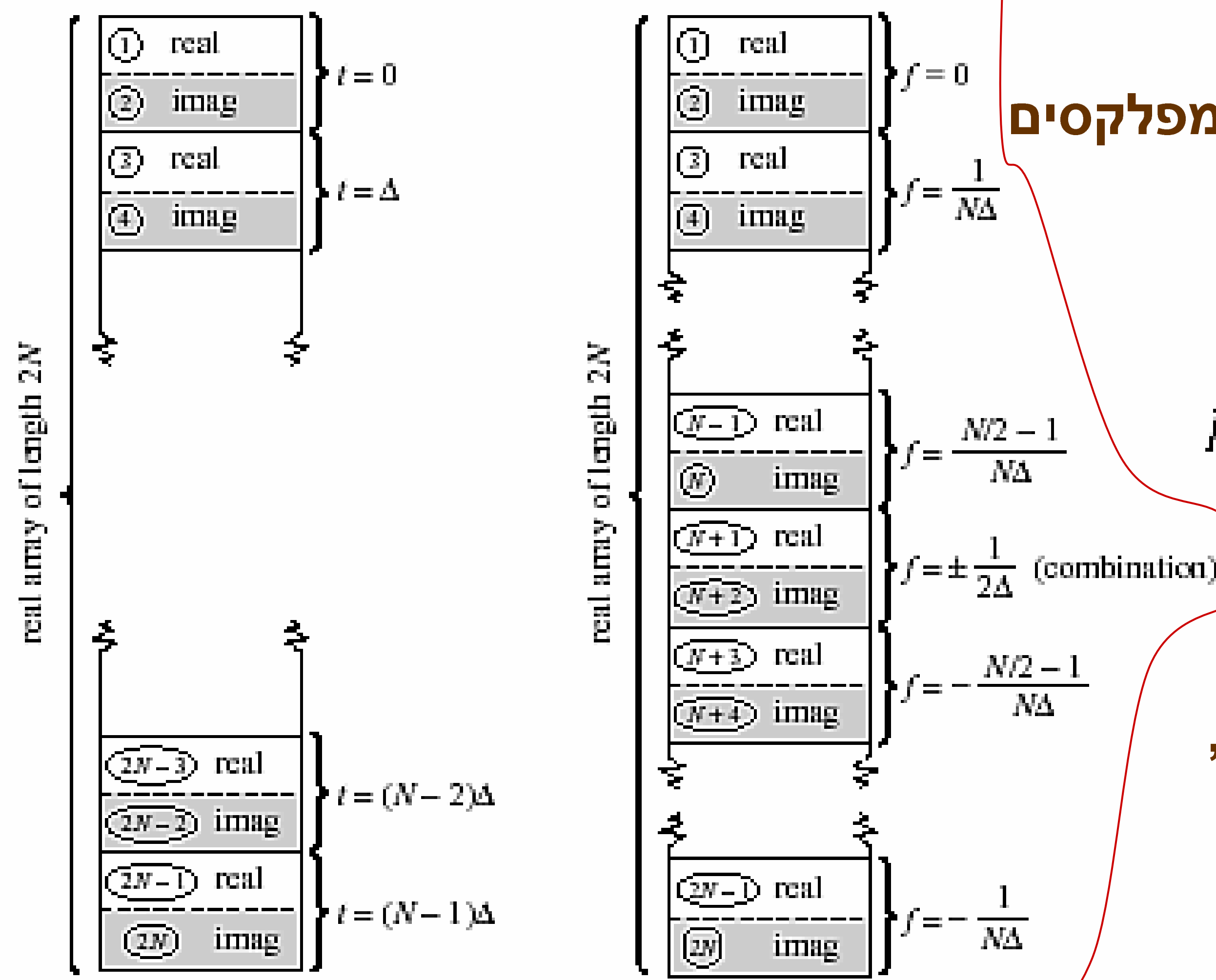
h: רכיב ממשי ורכיב דמיוני, לסירוגין, לפי הסדר:

$$h_k \equiv h(t_k), \qquad t_k \equiv k\Delta$$

$$k = 0, 1, 2, \ldots, N-1$$

H: רכיב ממשי ורכיב דמיוני, לסירוגין, לפי הסדר:

$$f_n \equiv \frac{n}{N\Delta}$$



$n = 0, 1, \ldots, \frac{N}{2}$ **, ואז לא חוזרים על** $n = -\frac{N}{2}$ **שהוא כמו** $n = \frac{N}{2}$

אלא ממשיכים: $n = -\frac{N}{2}+1, -\frac{N}{2}+2, \ldots, -1$

למשל N=8: $n = 0, 1, 2, 3, 4, -3, -2, -1$

טרנספורם פוריה : סיכום

הבעיה: חישוב נומרי של טרנספורם פוריה בדיד (וההיפוך).

$$h_k \equiv h(t_k), \qquad t_k \equiv k\Delta, \qquad k = 0, 1, 2, \ldots, N-1$$

$$f_n \equiv \frac{n}{N\Delta}, \qquad n = -\frac{N}{2}, \ldots, \frac{N}{2}$$

$$H_n \equiv \sum_{k=0}^{N-1} h_k \, e^{2\pi i k n/N} \qquad h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n \, e^{-2\pi i k n/N}$$

השיטה: טרנספורם פוריה מהיר.

המהירות: $O(N \log_2 N)$ הפונקציה: **four1**

התדירות הקריטית: $f_c \equiv \dfrac{1}{2\Delta}$ חשובה בגלל משפט הדגימה ותופעת התרגום המוטעה.

Ordinary Differential Equations
משוואות דיפרנציאליות רגילות

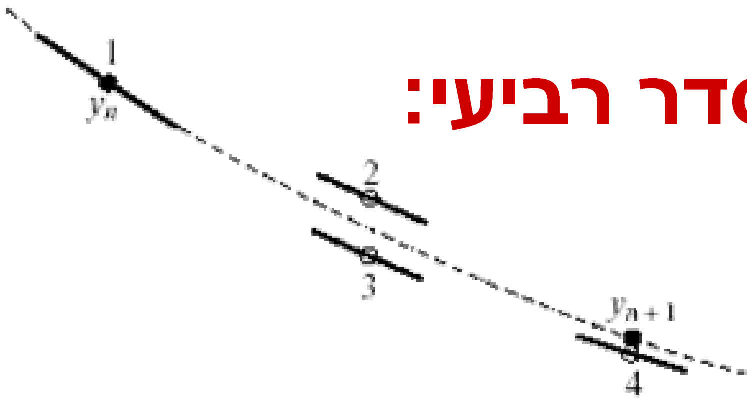
<u>תזכורת:</u> דוגמאות אנאליטיות (נפילה, צמיחה)

$$\frac{dy}{dt} + \alpha y = 0 \quad \Rightarrow \quad y = y_0 \exp(-\alpha t) \qquad \text{The } \textit{Decay} \text{ equation}$$

$$\frac{dy}{dt} - \alpha y = 0 \quad \Rightarrow \quad y = y_0 \exp(+\alpha t) \qquad \text{The } \textit{Growth} \text{ equation}$$

עכשיו: שקפים 1-7 על הלוח

שיטת Runge-Kutta מסדר רביעי:



<u>ניתן להוכיח את הנוסחא הבאה:</u>

תחילת הצעד:

$$k_1 = h\,f(x_n, y_n)$$

עד האמצע:

$$k_2 = h\,f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$$

שוב עד האמצע:

$$k_3 = h\,f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2)$$

הצעד המלא:

$$k_4 = h\,f(x_n + h, y_n + k_3)$$

$$y_{n+1} = y_n + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4 + O(h^5)$$

עכשיו: שקפים 8-9 על הלוח

משוואות דיפרנציאליות רגילות: סיכום

הבעיה: פתרון נומרי של מערכת משוואות:

$$\frac{dy_i(x)}{dx} = f_i(x, y_1, \ldots, y_N), \qquad i = 1, \ldots, N$$

שיטה 1: שיטת אוילר.

$$y(x_{n+1}) = y(x_n) + hf(x_n, y_n) + O(h^2)$$

שיטה 2: שיטת Runge-Kutta. מסדר שני:

$$\kappa_1 = hf(x_n, y_n)$$

$$\kappa_2 = hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}\kappa_1)$$

$$y_{n+1} = y_n + \kappa_2 + O(h^3)$$

מסדר רביעי:

$$y_{n+1} = y_n + O(h^5)$$

ניתן להעריך את השגיאה

ולשפר את הדיוק לסדר חמישי.

משוואות דיפרנציאליות חלקיות
(Partial Differential Equations)

משוואות היפרבוליות: גלים

$$\frac{\partial^2 u}{\partial t^2} = v^2 \frac{\partial^2 u}{\partial x^2}$$

משוואות פרבוליות: דיפוזיה

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial u}{\partial x}\right)$$

משוואות אליפטיות:
Poisson, Laplace

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \rho(x, y)$$