Computer Science CSC373H | September 21, 2022
St. George Campus | University of Toronto

Solutions for Homework Assignment #1

Answer to Question 1.

a.

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

b. Let $\vec{v}$ be a column vector of length $n = 2^k$ for some $k \geq 1$. We want to compute the product $M_k\,\vec{v}$. Note that the length $n = 2^k$ of $\vec{v}$ tell us which DeSica matrix M_k must be multiplied by $\vec{v}$. So we don't need M_k as part of our input, getting $\vec{v}$ as the input suffices.

Let $\vec{v}_1$ and $\vec{v}_2$ be the first and second half of $\vec{v}$, respectively. (So, each of $\vec{v}_1$ and $\vec{v}_2$ has length $n/2 = 2^{k-1}$.) Then, by the definition of M_k we have:

$$M_k\vec{v} = \begin{bmatrix} M_{k-1} & M_{k-1} \\ M_{k-1} & -M_{k-1} \end{bmatrix} \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \end{bmatrix} = \begin{bmatrix} M_{k-1}\vec{v}_1 + M_{k-1}\vec{v}_2 \\ M_{k-1}\vec{v}_1 - M_{k-1}\vec{v}_2 \end{bmatrix} = \begin{bmatrix} \vec{u}_1 \\ \vec{u}_2 \end{bmatrix}$$

So, we can compute $M_k\vec{v}$ by computing $M_{k-1}\vec{v}_1$ and $M_{k-1}\vec{v}_2$ (recursively) and doing a linear number of scalar additions or subtraction. In other words, we can multiply the $n \times n$ DeSica matrix by a n-vector $\vec{v}$, by performing two multiplications of the $n/2 \times n/2$ DeSica matrix by $n/2$-vectors ($\vec{v}_1$ and $\vec{v}_2$) and n scalar additions or subtractions.

DS-MULT($\vec{v}$)
1 {* assumes that $\vec{v}$ is a column vector of length $n = 2^k$ for some $k \geq 0$ and recursively computes $M_k\vec{v}$ *}
2 $n := |\vec{v}|$
3 **if** $n = 1$ **then return** $\vec{v}$
4 $\vec{v}_1 :=$ the first $n/2$ elements of $\vec{v}$; $\vec{v}_2 :=$ the last $n/2$ elements of $\vec{v}$
5 {* recursively compute $\vec{w}_1 = M_{k-1}\vec{v}_1$ and $\vec{w}_2 = M_{k-1}\vec{v}_2$ *}
6 $\vec{w}_1 :=$ DS-MULT($\vec{v}_1$); $\vec{w}_2 :=$ DS-MULT($\vec{v}_2$)
7 $\vec{u}_1 := \vec{w}_1 + \vec{w}_2$; $\vec{u}_2 := \vec{w}_1 - \vec{w}_2$
8 $\vec{u} := \begin{bmatrix} \vec{u}_1 \\ \vec{u}_2 \end{bmatrix}$
9 **return** $\vec{u}$

c. Since dividing and combining the solutions of the two subproblems of size $n/2$ takes $O(n)$ time (see lines 4, 7 and 8), the recurrence for the running time of the algorithm is:

$$T(n) = \begin{cases} 2T(n/2) + cn, & \text{if } n > 1 \\ c, & \text{if } n = 1 \end{cases}$$

for some constant c. By the Master Theorem (where $a = 2$, $b = 2$ and $d = 1$, so $a = b^d$):

$$T(n) = O(n \log n)$$