Performance evaluation: Arnoldi's algorithm

Nicolas Derumigny

ENS Lyon

29 Novembre 2017

1 $q_1 = \frac{q}{||q||}$
2 **for** *k=1 à m-1* **do**
3 $\quad w = Aq_k$
4 $\quad$ **for** *j=1 à k* **do**
5 $\quad\quad h_{j,k} = \langle w, q_0 \rangle$
6 $\quad\quad w = w - h_{j,k} \cdot q_j$
7 $\quad h_{k+1,k} = ||w||_2$
8 $\quad q_{k+1} = \frac{w}{h_{k+1,k}}$

Algorithm 1: Arnoldi's algorithm

How to measure the execution time?

- Use `time.h`

```
#include <time.h>

int main(){
        clock_t start, end;
        start = clock();
        //program here
        end = clock();
        printf("%f\n", (end-start)/CLOCKS_PER_SEC));
}
```

Complexity and tests

- Time: The matrix-vector product dominates $\Rightarrow O(n^2)$
- Space: The storage of A dominates $\Rightarrow O(n^2)$

The multiplications have been tested on

- $\begin{pmatrix} 1 \\ 1 \end{pmatrix}^2 = 2$
- $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

i5-2520M

- Sandy Bridge
- 3.5 GHz
- Hyperthreaded Dual-core
- $M = 80$

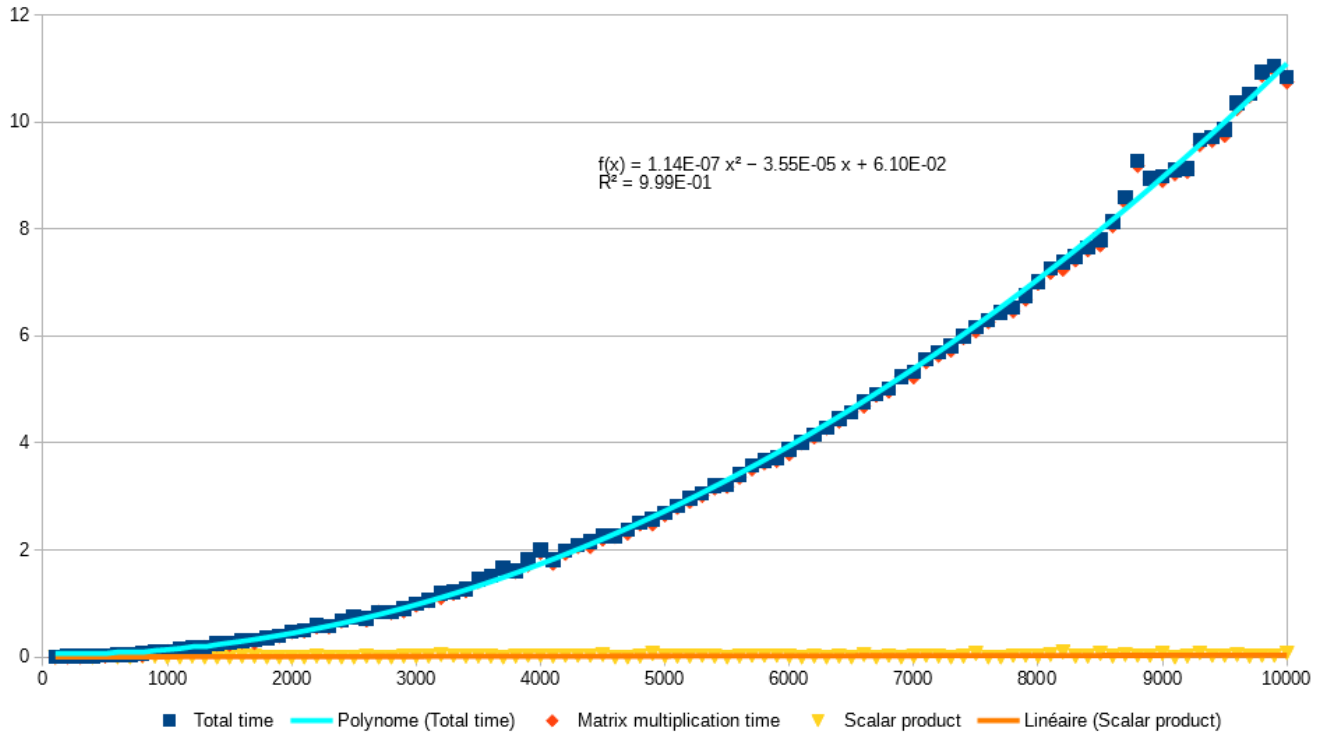

Figure: Performance evaluation on the i5-2520M: execution time $= f(\text{matrix size})$ and quadratic model $time = f(size) = A.size^2 + B.size + C$