

Thread	ms	
1	4656	
2	2245	
3	1655	
4	1234	
5	1093	
6	946	
7	847	
8	790	
9	706	
10	749	
11	681	
12	561	
13	610	
14	537	
15	553	
16	510	
<b>17</b>	492	
<b>18</b>	523	
<b>19</b>	483	
<b>20</b>	492	
<b>21</b>	484	
<b>22</b>	491	
<b>23</b>	501	

10 physical cores and 16 threads(intel hyper-threading)

—

**Decrescita iniziale significativa:**

L'aumento del numero di thread da 1 a circa 8 comporta un miglioramento sostanziale delle prestazioni. Si osserva un passaggio da 4656 ms a circa 790 ms, indicando un'efficace

parallelizzazione dei task su più core.

#### **Diminuzione del beneficio marginale:**

Tra 9 e 16 thread, i guadagni in termini di tempo di esecuzione diventano meno marcati. Il tempo scende ancora, ma in maniera meno accentuata. Questo comportamento è tipico quando si raggiunge la saturazione dei core fisici e si inizia a sfruttare l'hyper-threading.

#### **Regione di saturazione (oltre 16 thread):**

Oltre i 16 thread (limite dell'hyper-threading su CPU con 10 core fisici e 16 thread logici), le prestazioni si stabilizzano. Il tempo si assesta tra 483 e 523 ms, con variazioni minime, indicando che ulteriori thread non apportano miglioramenti significativi. In alcuni casi, si osservano addirittura lievi peggioramenti dovuti al sovraccarico di gestione dei thread (content switch).

#### **Anomalia locale (thread 10 e 13):**

Si notano alcune fluttuazioni non monotone (ad esempio, il tempo a 10 thread è superiore rispetto a 9), che possono essere attribuite a variabili esterne come il carico del sistema operativo, scheduling, o interferenze da altri processi in esecuzione.

#### **Andamento generale:**

- All'aumentare dei thread, il tempo di esecuzione diminuisce rapidamente all'inizio.
- Il miglioramento rallenta attorno ai **10 thread**, corrispondenti ai **core fisici** della CPU.
- Intorno ai **16 thread**, corrispondenti ai **thread logici** abilitati tramite **Hyper-Threading**, il tempo continua a migliorare, ma con benefici sempre più marginali.
- Oltre i 16 thread, l'andamento si stabilizza: i tempi oscillano attorno ai **500 ms**, senza più un guadagno significativo.

#### **Aumento ulteriore dei thread:**

##### **Aumento dell'overhead di gestione:**

- Il sistema operativo deve gestire più thread simultaneamente tramite **time-slicing**, causando:
  - **Switching costante** tra thread su core già occupati.
  - Aumento dell'**overhead del context switching**.
  - Degrado delle performance dovuto alla **mancanza di locality**

##### **Possibile peggioramento delle prestazioni:**

- In assenza di sufficiente parallelismo nel carico di lavoro o con risorse condivise (RAM, I/O), thread aggiuntivi non solo **non aiutano**, ma **rallentano l'esecuzione**.
- Questo effetto si osserva in molti benchmark reali: il tempo di esecuzione **oscilla o peggiora** leggermente oltre un certo numero di thread.

