Sustainability

Objective: Reduce the carbon footprint of computing applications

Ideas

- Define application specific standards for data packing so serving is as efficient as possible

  - A lot of data reading and serving tools (e.g., databases, file formats, etc.) aim to be general purpose in order to increase the number of users. This approach misses the opportunity to optimize data accessing for applications that don't need the full set of features available in these tools.

  - From the Facebook AI sustainability paper: "... application-level caching improves power efficiency by $6.7\times$".

- Control the allocation of resources so that strong scaling is fixed at the strong-scale limit[1], thus minimizing the overhead impact of distributed computing infrastructure management

  - Strong scaling allows an application to add more computing resources that proportionally reduce the execution time. The parallelization of the application adds some overhead which in turns affects the total potential speedup. In many cases this cannot be avoided, and so limitting the number of resources given to the system can reduce the power consumption of the application that goes into overhead.

  - On a different direction, add as many computing resources as needed so the execution time and power consumption are minimized.

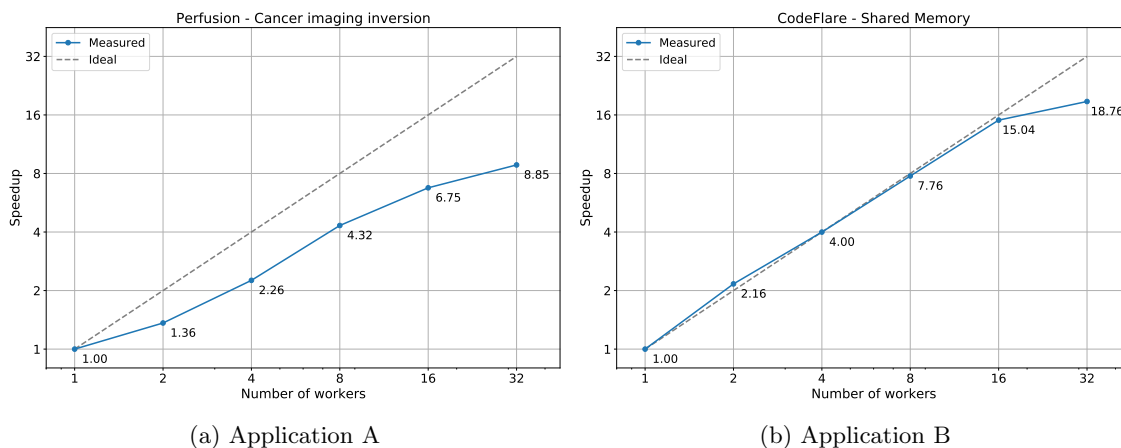(a) Application A   (b) Application B

Figure 1: Strong scaling of two **very** different applications exhibiting different performance. Application B strong scales well up to 16 processors while Aplication A is not using the resources efficiently resulting in bad scaling.

- Load distribution accross centers operating in different time zones (some data-centers won't be fully utilized at night)

---

[1] *Strong-scale limit:* Scaling point at which adding more processors doesn't reduce the execution time in the same proportion.