VOLUNTEER COMPUTING: AN ENERGY-CONSUMPTION PERSPECTIVE

Ana-Lucia Varbanescu

University of Twente, NL

Energy concerns around computing

- Top 10 videos on YouTube* consumed as much as 600-700 EU persons per year (or about 400 North America persons)
- Training Alpha-Zero for a new game consumes as much as 100 EU persons per year
- A mid-size data-center alone consumes as much energy as a small town
  - And that is not considering purchasing and secondary operational costs (e.g., cooling)
- In 2019 Dutch datacenters consumed 3x more energy than the national railways
  - And consumption increased by 80% in 3 years
- The ICT sector is predicted to reach 21% of the global energy consumption by 2030

*https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos#Top_videos

Energy concerns around computing

- Top 10 videos on YouTube* consumed as much as 600-700 EU persons per year (or about 400 North America persons)

- Training Alpha-Zero for a new game consumes as much as 100 EU persons per year

- A mid-size data-center alone consumes as much energy as a small town

> There is an imperative need to reduce energy consumption and especially energy waste in computing.

- I
  - And consumption increased by 80% in 3 years

- The ICT sector is predicted to reach 21% of the global energy consumption by 2030

*https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos#Top_videos

Stakeholders

Developers and users

Improve the energy efficiency of their own codes, making use of algorithmic, programming, and hardware tools

Design and implement applications able to adapt to the available system resources

System integrators

Offer the right mix of resources for the application developers and system operators.

Include efficient hardware to enable different application mixes.

System operators

Ensure efficient scheduling of workloads on system resources.

Harvest energy where resources/systems are massively underutilized.

Improving energy efficiency

Raise awareness

- Monitor (energy) efficiency
- Quantify waste

Improve efficiency

- Improve applications for the systems at hand
- Improve systems* for the applications at hand
  - Better hardware
  - Efficient multi-tenancy, better scheduling
- Co-design applications and systems

Analysis

Modeling

Application optimization

Hardware/tech evolution

Efficient scheduling and resource sharing

??

*That includes hardware and system software

Multi-tenancy

Data center

- Pro:
  - Up-to-date HW and SW
  - Dedicated/stable resources
  - Fast computation & networking
  - Efficient scheduling
  - Job collocation
  - Efficient optimizations for sustainability
- Con:
  - Low per-application utilization
  - Dedicated resources

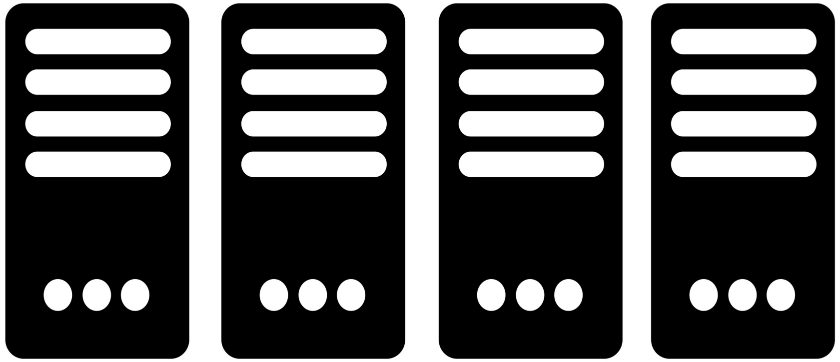
Volunteer

- Pro:
  - Existing resources
  - Built-in OS-based multi-tenancy
- Con:
  - Slow(er) computation and networking
  - Difficult to reserve/account for resources
  - Reduced fault-tolerance and reliability
    - Expensive redundancy
  - Consumer-grade machines

Execution model

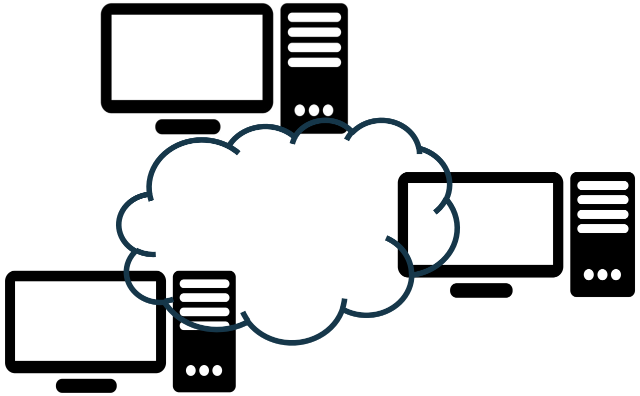
Data center

- Tightly coupled
- Faster
- Less communication
- Dedicated resources => ?? energy



Volunteer

- Loosely-coupled
- Slower
- More communication
- Spare resources => ?? energy



Tecnologia Vectors by Vecteezy, https://www.vecteezy.com/free-vector/tecnologia
Computer Vectors by Vecteezy, https://www.vecteezy.com/free-vector/computer

Per task energy

Data center

$E_D (i) = T_D(i) * P_D$

$T_D (i) = FLOPS(Task_i) / Peak(CPU_D)$

$P_D = TDP$

Volunteer

$E_V(i) = (T_V(i) + T_{RV}(i)) * P_V$

$T_V(i) = FLOPS(Task_i) / Avail(CPU_V)$

$T_{RV}(i)$ = overhead to transfer results

$P_V = TDP – P_{idle}$

$$E_D(i) / E_V(i) = T_D(i) / (T_V(i) + T_{RV}(i)) \times (TDP / (TDP-P_{idle})) \sim Avail/Peak \times (1 + P_{idle}/P_{used})$$

Per task energy

Data center

$E_D(i) = T_D(i) * P_D$

$T_D(i) = FLOPS(Task_i) / Peak(CPU_D)$

$P_D = TDP$

Volunteer

$E_V(i) = (T_V(i) + T_{RV}(i)) * P_V$

$T_V(i) = FLOPS(Task_i) / Avail(CPU_V)$

$T_{RV}(i)$ = overhead to transfer results

$P_V = TDP – P_{idle}$

Accounts for using the extra power when the resource is not idle anymore

Accounts for user vs. data-center compute availability

$$E_D(i) / E_V(i) = T_D(i) / (T_V(i) + T_{RV}(i)) \times (TDP / (TDP-P_{idle})) \sim Avail/Peak \times (1 + P_{idle}/P_{used})$$

Per task energy: DC or VC ?

- $E_D(i) / E_V(i) \sim Avail/Peak \times (1 + idle/used)$

Proportionally slower than DC!

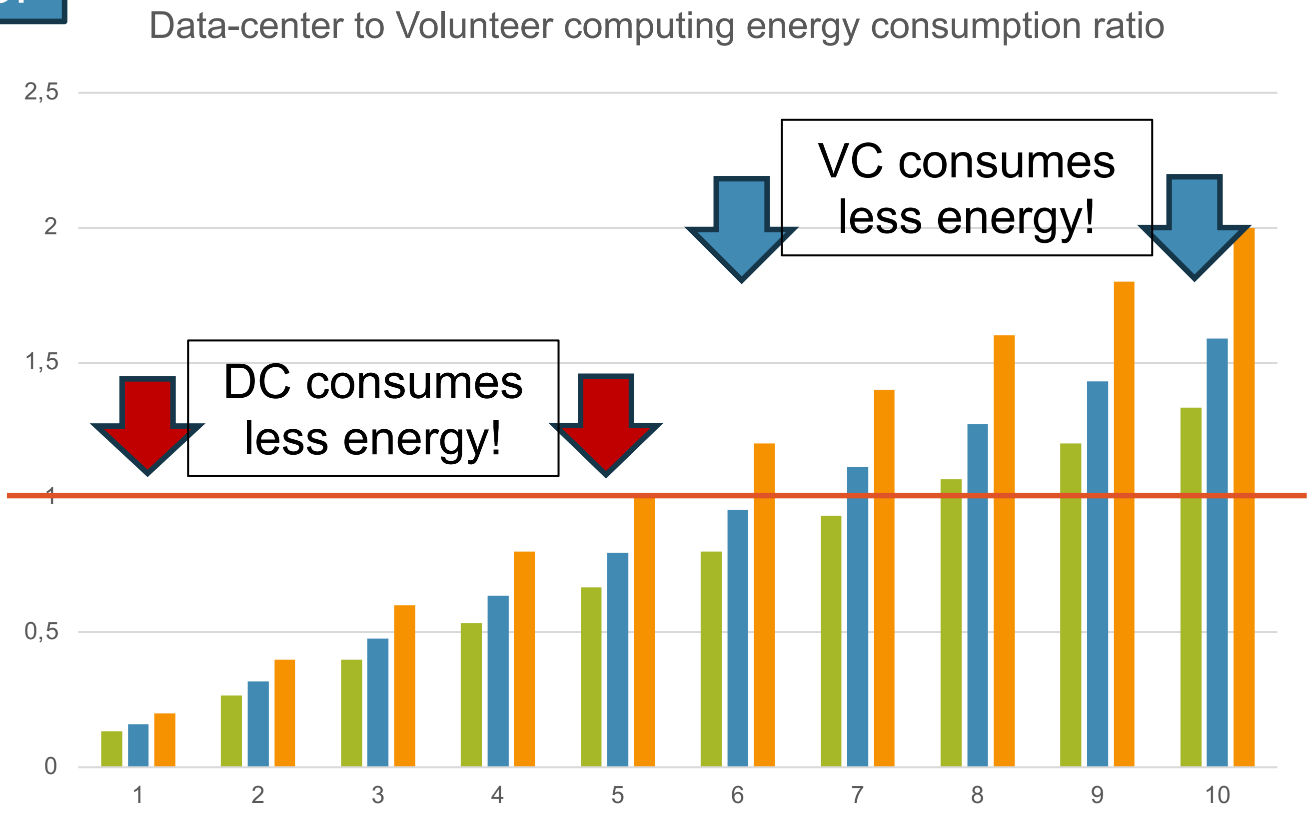
- Assume …
  - Volunteer availability: 10-100%
  - Machines M1 – M3
    - 0,33 idle/used (green)
    - 0,58 idle/used (blue)
    - 1,00 idle/used (orange)

Per task energy: DC or VC ?

- $E_D(i) / E_V(i) \sim$ Avail/Peak x (1 + idle/used)

Proportionally slower than DC!

- Assume …
  - Volunteer availability: 10-100%
  - Machines M1 – M3
    - 0,33 idle/used (green)
    - 0,58 idle/used (blue)
    - 1,00 idle/used (orange)



Per task energy: DC or VC ?

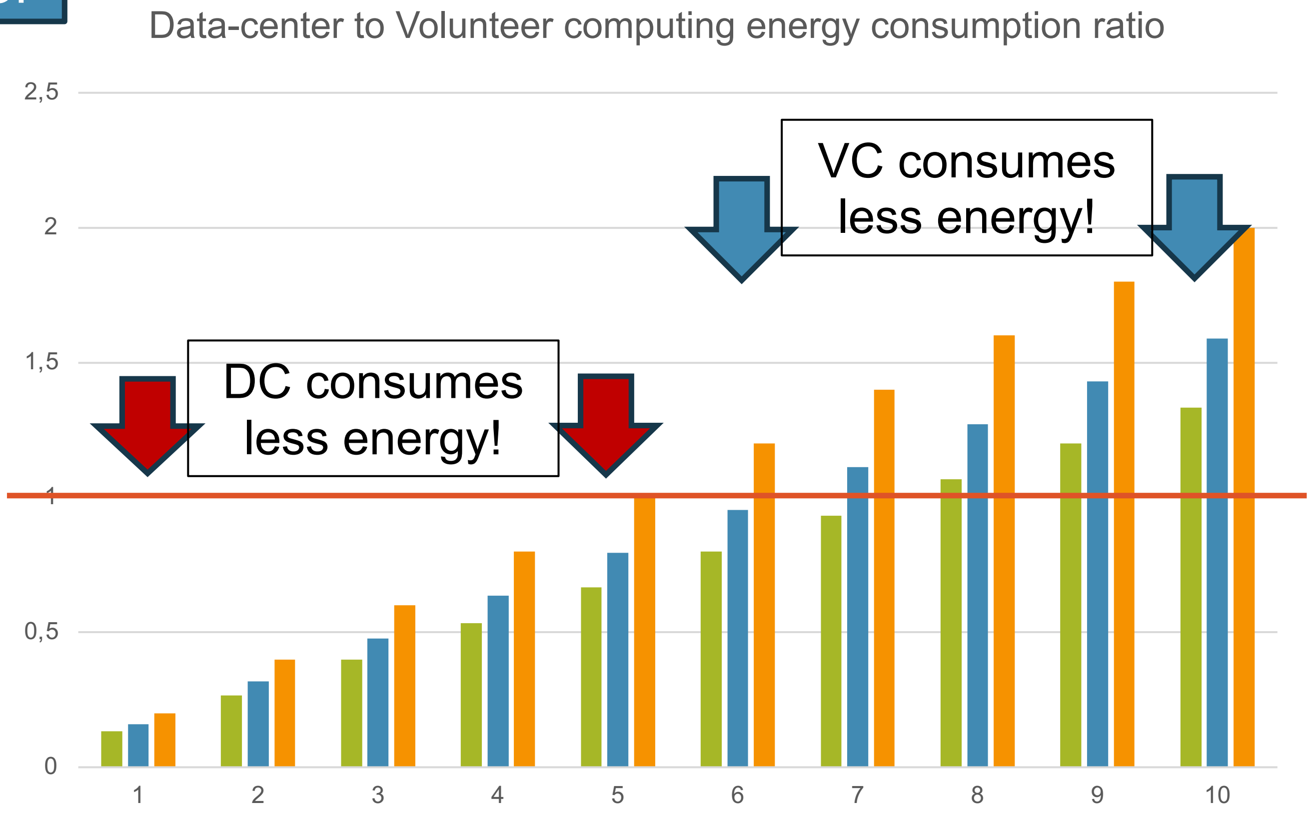
- $E_D(i) / E_V(i) \sim$ Avail/Peak x (1 + idle/used)

Proportionally slower than DC!

- Assume …
  - Volunteer availability: 10-100%
  - Machines M1 – M3
    - 0,33 idle/used (green)
    - 0,58 idle/used (blue)
    - 1,00 idle/used (orange)

There are opportunities to be more energy efficient per task for decentralized computing!

It ultimately depends on the type of machines and available cycles from the users…



What about the full application?

Data center

$E_D = \text{sum}\ (E_D(i))$

... But here we need to take into account the TCO, especially for on-prem hardware …

Volunteer

$$E_V = \text{sum}\ (E_{task}(i) \times R + E_{selection}(i) + E_{comm}(i) + E_{scheduler}(i))$$

Energy gains **also** depend on how efficient we are on redundancy, communication, scheduling

TCO = total cost of operation

What about sustainability?

Data center

Pro:

- Additional mechanisms for in-time and in-space scheduling => better energy mix

Con:

- Total cost of ownership
- Additional concerns regarding infrastructure and cooling

Volunteer

Pro:

- Distributed infrastructure => high probability for better energy mix
  - Implicit in-space scheduling
- Reduces compute waste to a minimum
- Default in-time scheduling

Con:

- Additional and redundant computations

What next ?

- Collect more data
  - About the machines
  - About the user availabilities
  - About redundancy, scheduling and networking costs
- Build simulators/digital twins for such systems
  - There exist data-center simulations
  - There exist Edge/Fog computing simulations
- Quantify the reduction in compute waste for volunteer computing
- Assess the change in software to account for …
  - Mobile computing
  - Data movement costs

We can create together the first model(s) to estimate energy gain (or reduction in energy waste) for volunteer computing !!

Take home message
to-the-office

- Volunteer computing can be a feasible alternative for sustainability in scientific computing.

- Its success depends on …
  - Software efficiency
  - User contributions in terms of systems and time/cycles

- Better models/more data is needed for more accurate models …
  - But the outlook is positive!