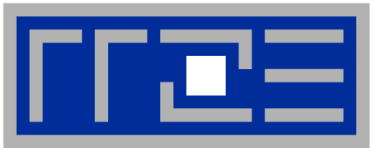
Assignment 5 – Task 1
Peak performance of Xeon Phi/Nvidia Volta



- Xeon Phi “Knights Landing”
  - 64 cores
  - f=1.3 GHz
  - AVX-512 instruction set (512-bit registers, 2 full-width FMA instructions per cycle)

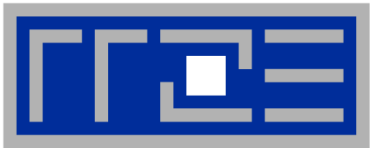
$$P_{peak} = 64 \times 2\,\frac{instr}{instr} \times 2\,\frac{instr}{cy} \times 16\,\frac{flop}{instr} \times 1.3\,\frac{Gcy}{s} = 5{,}325\ \text{GFlops/s}$$

cores | n_{FMA} | n_{super} | n_{SIMD} | f

- Nvidia Volta V100
  - 80 SM units (streaming multiprocessors), 64 SP “cores” w/ 1 FMA each
  - f=1.4 GHz

$$P_{peak} = 80 \times 2\,\frac{instr}{instr} \times 1\,\frac{instr}{cy} \times 64\,\frac{flop}{instr} \times 1.4\,\frac{Gcy}{s} = 14{,}336\ \text{GFlops/s}$$

Assignment 5 – Task 2
A simple power model for multicore chips $W(n, f)$



- W_d Dynamic power consumption of a running core (see Assignment 4 – T3)
- Δv relative clock frequency change (see Assignment 4 – T3
- n cores used (of n_{max} available) (see Assignment 4 – T3)
- W_0 Baseline power consumption of the chip (all cores idle): $W(n = 0) = W_0$

→ **Power** $W(n, f) = W_0 + nW_d(1 + \Delta v)^3$

Assignment 4 – T3

- P_0 Performance of serial program @ $\Delta v = 0$
- P_{max} Maximum performance of parallel program
- P(n) Performance of parallel program on n cores
- T(n) Time to solution on n cores

→ **Time to solution** $T(n) = \frac{1}{\min(nP_0(1+\Delta v), P_{max})}$

- E(n) Energy to solution

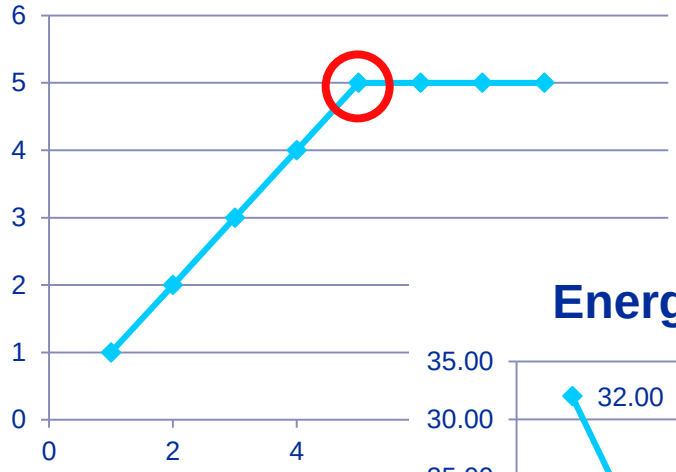
→ **Energy to solution** $E(n) = W(n, f)T(n) = \frac{W_0 + nW_d(1+\Delta v)^3}{\min(nP_0(1+\Delta v), P_{max})}$

(a) Minimal energy to solution

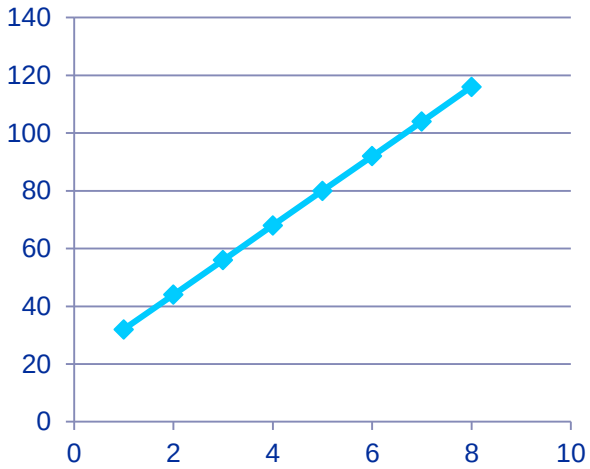
$$E(n) = \frac{W_0 + nW_d(1+\Delta\nu)^3}{\min(nP_0(1+\Delta\nu), P_{max})}$$

$$P_0 = 1\,\mathrm{s}^{-1}$$
$$P_{max} = 5\,\mathrm{s}^{-1}$$
$$W_0 = 20\,\mathrm{W}$$
$$W_d = 12\,\mathrm{W}$$
$$\Delta\nu = 0$$

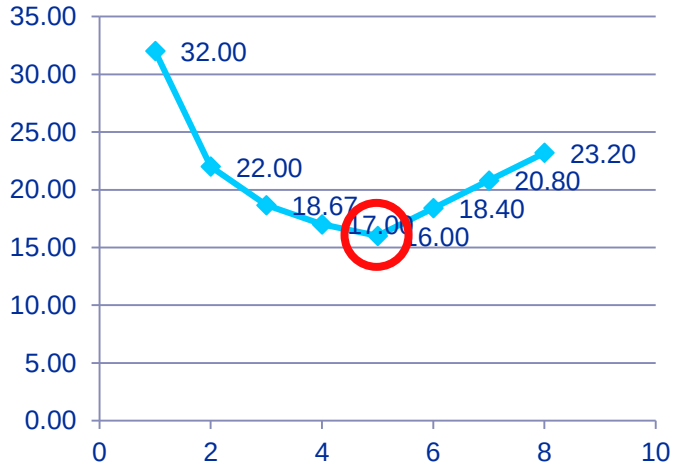
Performance



Power



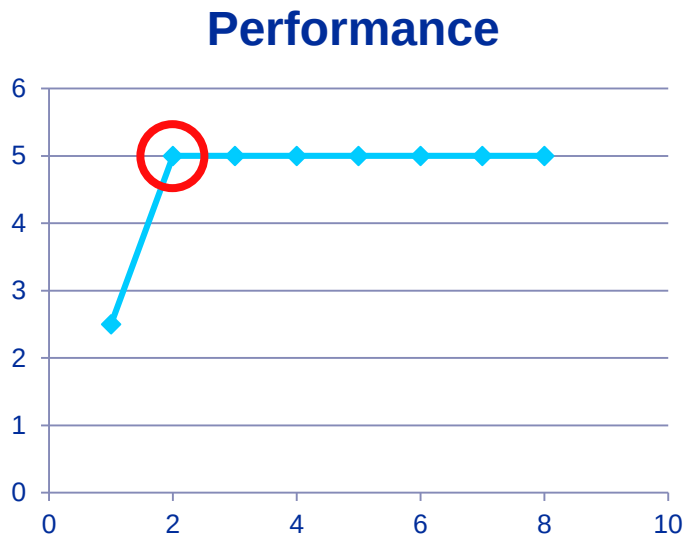
Energy to solution

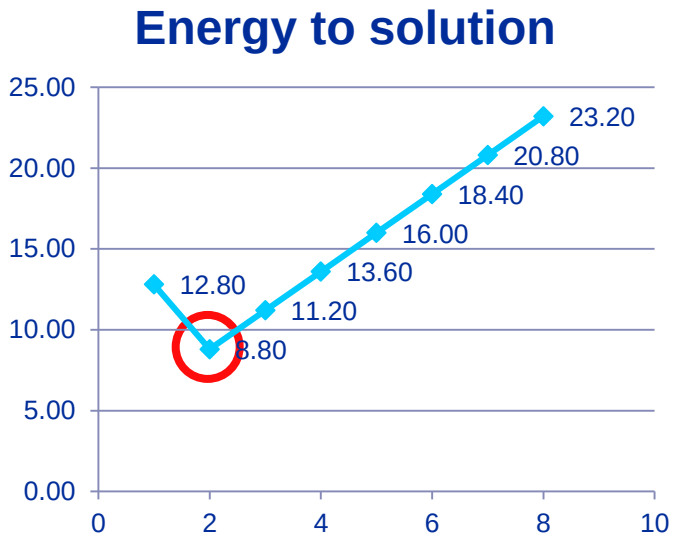


→ Minimal energy at the **saturation** point!

(b) Energy to solution: Increase serial performance

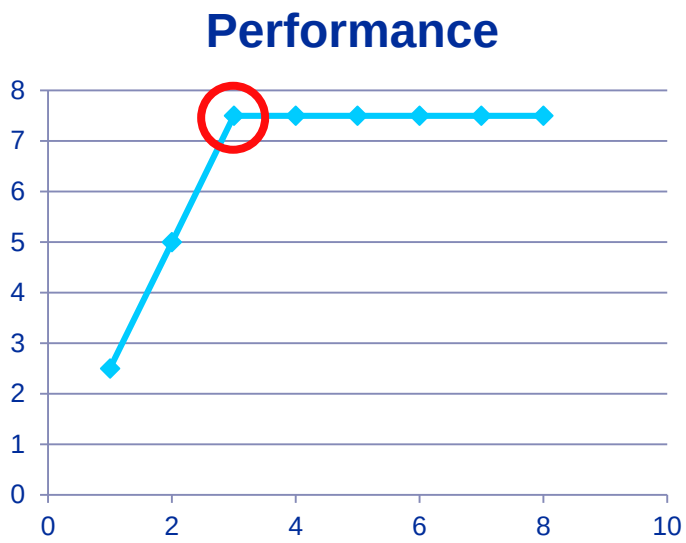
$$P_0 = 2.5\ \mathrm{s}^{-1}$$
$$P_{max} = 5\ \mathrm{s}^{-1}$$
$$W_0 = 20\ \mathrm{W}$$
$$W_d = 12\ \mathrm{W}$$
$$\Delta\nu = 0$$



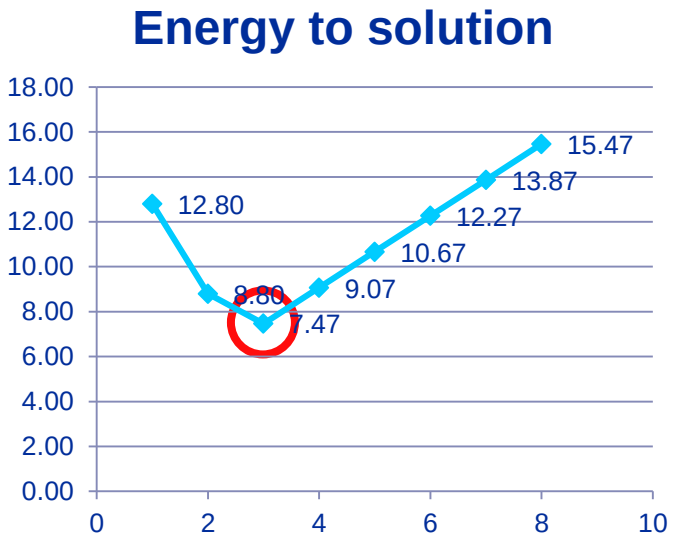


→ Smaller energy to solution at *smaller* core count with faster serial code!

(c) Energy to solution: Increase saturated performance



$$P_0 = 2.5\ \mathrm{s}^{-1}$$
$$P_{max} = 7.5\ \mathrm{s}^{-1}$$
$$W_0 = 20\ \mathrm{W}$$
$$W_d = 12\ \mathrm{W}$$
$$\Delta\nu = 0$$



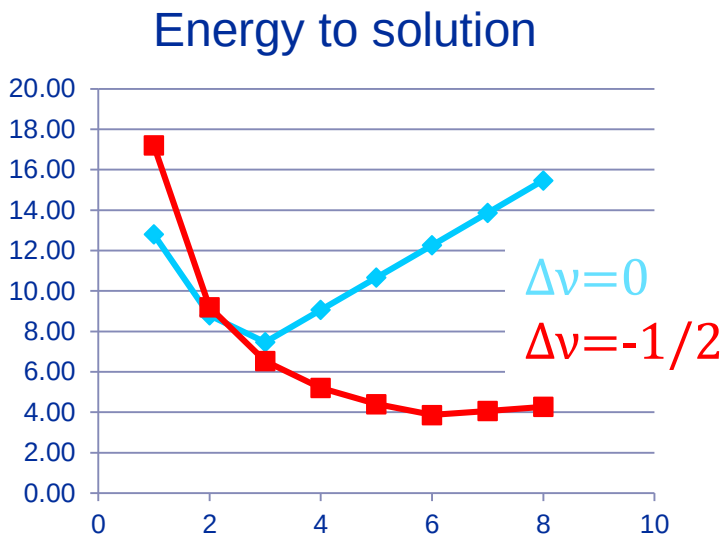
→ Smaller energy to solution at *larger* core count with larger saturated performance!

(d) Energy to solution: Dependence on clock speed

- Prerequisite: We do not want to sacrifice saturated performance → $nP_0 \geq P_{\max}$
- For small W_0, expect dominant impact of numerator in

$$E(t) = \frac{W_0 + nW_d(1+\Delta\nu)^3}{\min(nP_0(1+\Delta\nu), P_{max})}$$

- Consequence: Turn down clock speed so that $P_{\max}$ is reached as late as possible



Assignment 5 –Task 3: Vector triad on Emmy

```
double precision, dimension(:), allocatable :: A,B,C,D

allocate(A(1:N),B(1:N),C(1:N),D(1:N))
A=1.d0; B=A; C=A; D=A
!$OMP PARALLEL private(j)
do j=1,NITER
!$OMP PARALLEL DO
  do i=1,N
    A(i) = B(i) + C(i) * D(i)
  enddo
!$OMP END PARALLEL DO
  if(.something.that.is.never.true.) then
    call dummy(A,B,C,D)
  endif
enddo
!$OMP END PARALLEL
```

Outer parallel

Assignment 5 –Task 3:
Vector triad on Emmy (1 socket @ 2.2 GHz)