

# Energy Usage for Parallel Haskell Programs

---

**Kevin Hammond**

Hylomorph Solutions

WG2.11, Salem, May 2nd, 2019

Work undertaken by:  
Yasir Alguwaifli  
University of St Andrews  
In: Proc. Lambda Days 2019  
Submitted to CC&PE

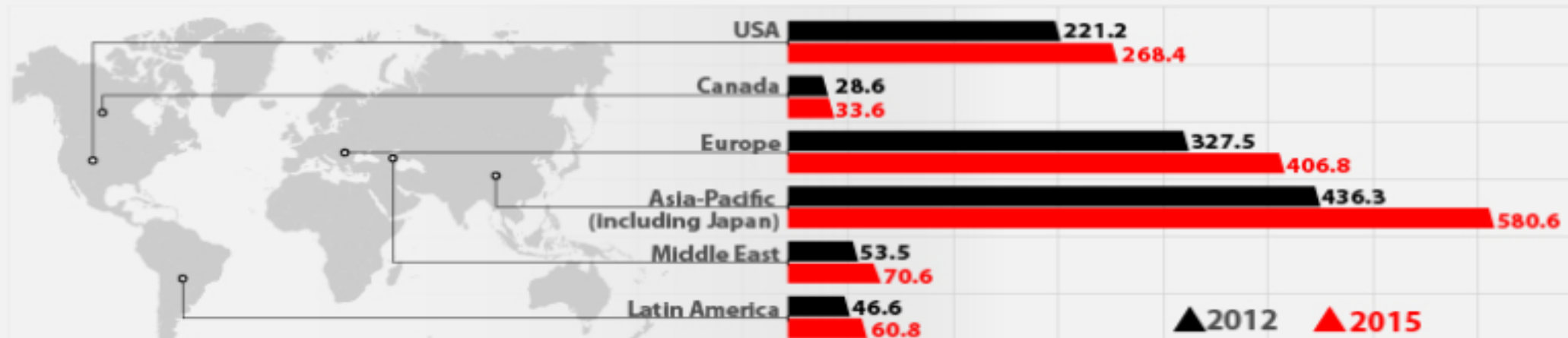


# Multicore Momentum

|                        |                  |
|------------------------|------------------|
| <b>Cloud Computing</b> | <b>IoT</b>       |
| Performance            | Embedded Systems |
| Multimedia             |                  |

## Growing Sales of Consumer Electronics Spur Opportunities for Multi-Core Processors

*Global Revenues for Consumer Electronics (In US\$ Billion) for Years 2012 & 2015*





# From Multicore to Manycore



Intel Xeon Phi 7290: **72** x86 cores (at 1.5GHz), 245W  
Aimed at supercomputing applications

# It's not just about large systems

- **Even mobile phones are multicore**
  - Samsung Exynos 5 Octa has 8 cores, 4 of which are Dark
- **Performance/energy tradeoffs mean systems will be increasingly parallel**
- **Even embedded systems are becoming multicore and heterogeneous**
  - NVidia Tegra TX2 has 6 ARM CPU cores and 256-core Pascal GPU
- **If we don't solve the multicore/manycore challenge, then no other computing advances will matter!**



ALL Future Programming will be Parallel!

# Main Research Questions

**Q1: Can we relate energy usage to easily obtained metrics?**

**Q2: Is it possible to build good energy models for parallel Haskell programs?**

# Methodology

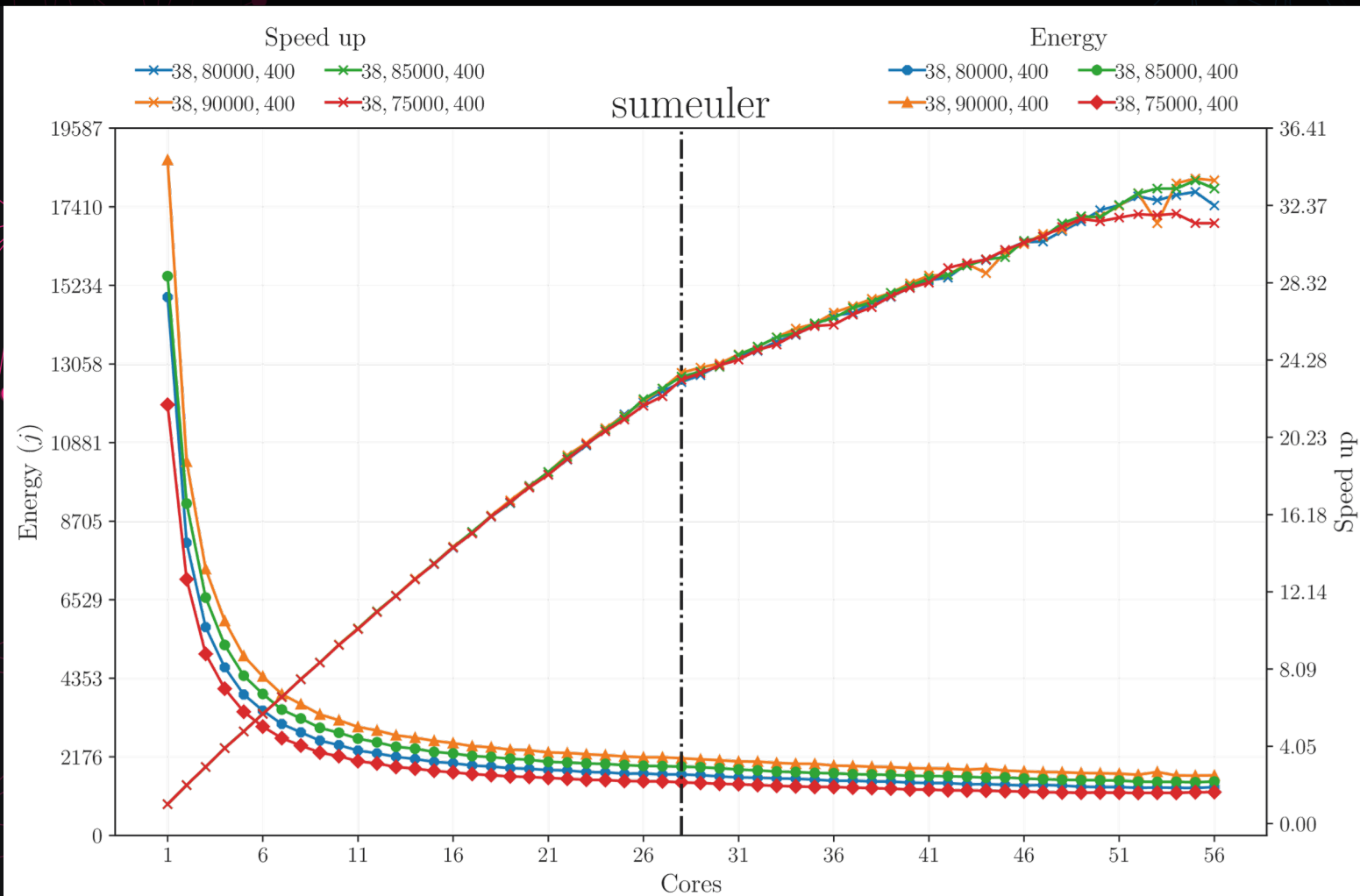
- Consider *scalable* parallel NoFib programs
  - Euler Summation; DFT (Parallel Map); N-Queens; N-Body; Matrix Multiplication; Parallel Fibonacci; Minimax.
- 4 different problem sizes for each program
- 10× samples for each data point
- Use the measured values to build a *predictive* model



## Experimental Setup

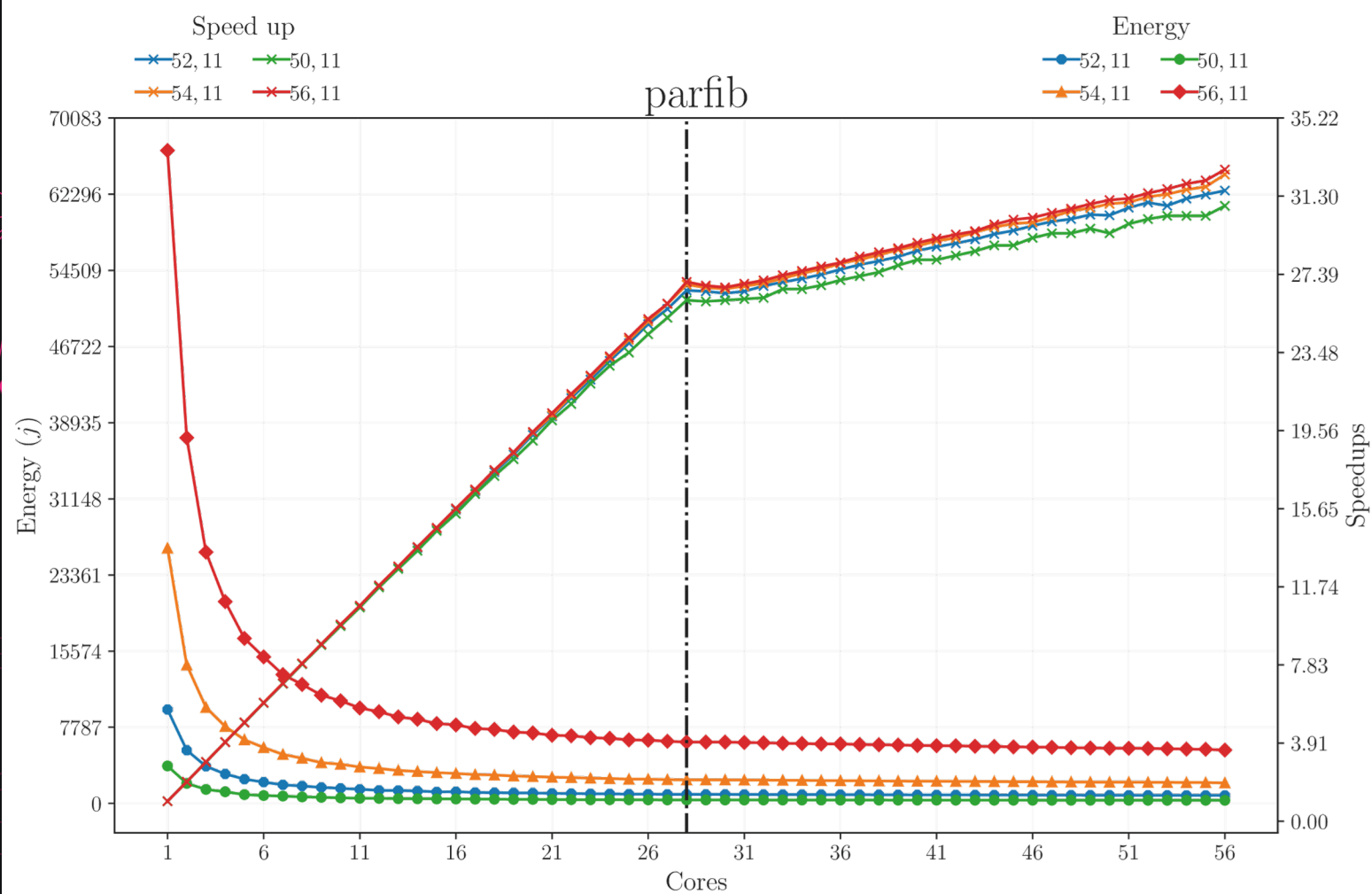
- Intel's Running Average Power Limit (RAPL) for energy sampling
- GHC 7.6.3 with -O2 & -feager-blackholing
- Red Hat 4.8.5-11 with kernel 3.10.0-514.21.2.el7.x86\_64
- 2×Intel Xeon E5-2690 @ 2.601 GHz, 28 Physical Cores
- Fixed Clock Frequency (DVFS disabled)

# Energy v Speedup: Euler Summation

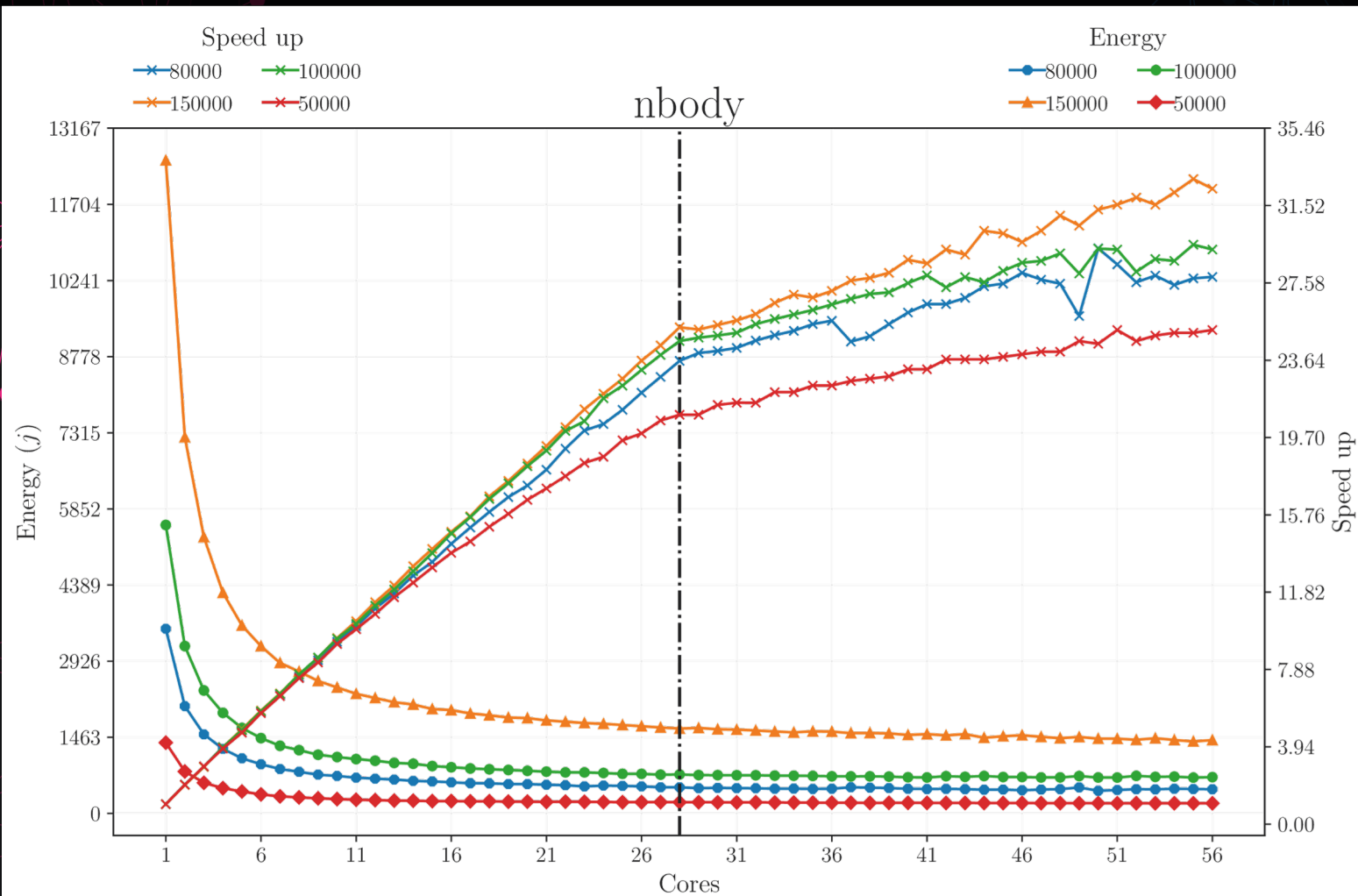




# Energy v Speedup: Parallel Fibonacci

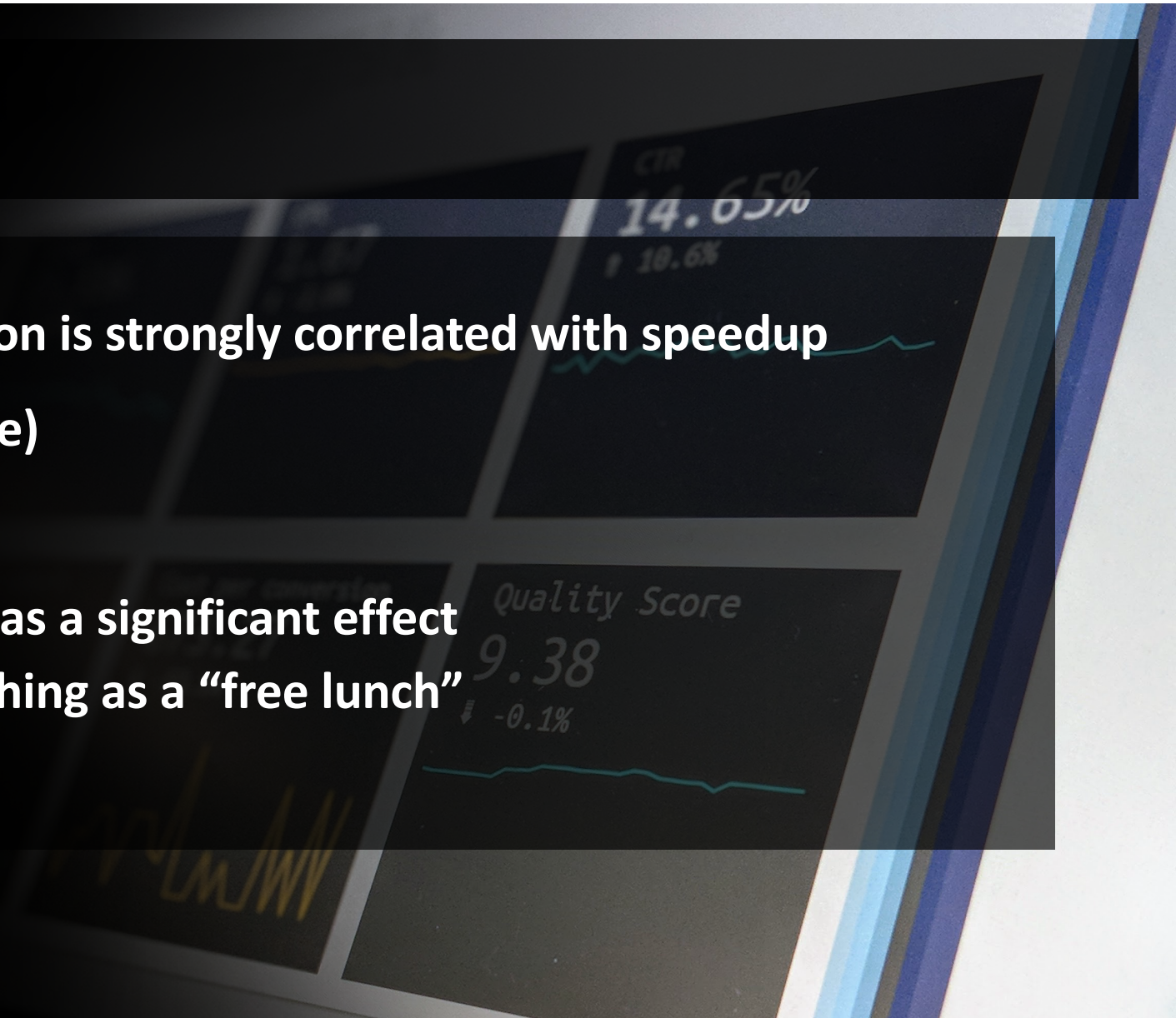


# Energy v Speedup: N-Body

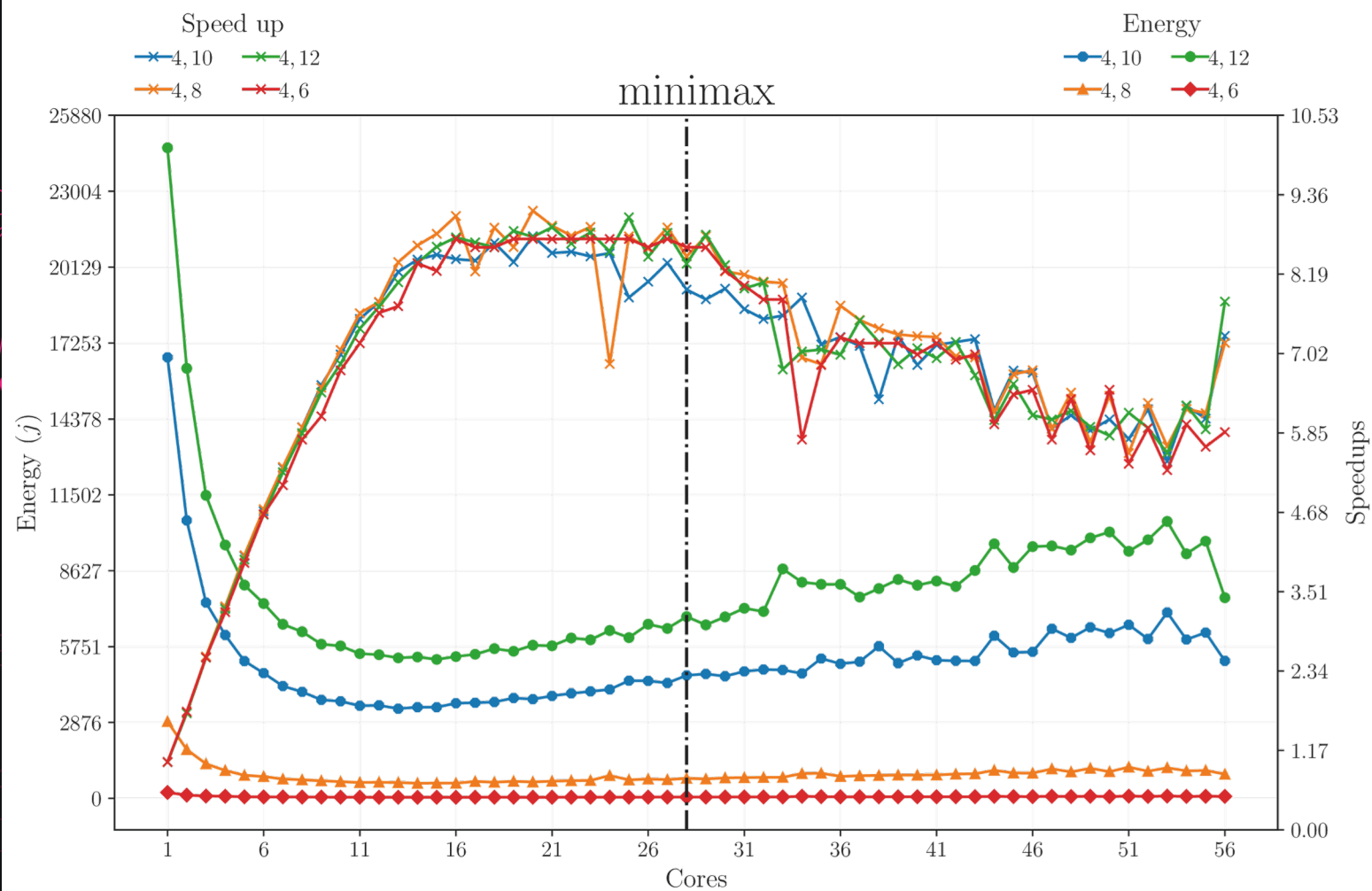


## Discussion

- Energy consumption is strongly correlated with speedup (i.e. execution time)
- Hyper-threading has a significant effect
  - There is such a thing as a “free lunch”



# Energy v Speedup: Minimax





## Discussion

- **Speedup is not the only factor governing energy usage**
  - Additional costs matter, e.g. execution time & GC



# Constructing an Energy Model

- **Apply Multi-variate regression**
  - Linear model:  $Y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_n x_{in}$
- **Following features considered:**
  - Productivity (Roughly = Speedup)
  - Garbage Collection (GC) Time
  - Execution ("Mutation") Time
  - Average Power Usage

CTR  
14.65%  
+ 10.6%

Quality Score  
9.38  
↓ -0.1%

# Constructing an Energy Model

- **Apply Multi-variate regression**
  - Linear model:  $Y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_n x_{in}$
- **Following features considered:**
  - Productivity (Roughly = Speedup)
  - Garbage Collection (GC) Time
  - Execution ("Mutation") Time
  - Average Power Usage
- $E = m c^2$

CTR  
14.65%  
+ 10.6%

Quality Score  
9.38  
↓ -0.1%



# Constructing an Energy Model

- **Apply Multi-variate regression**
  - Linear model:  $Y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_n x_{in}$
- **Following features considered:**
  - Productivity (Parallel Utilisation)
  - Garbage Collection (GC) Time
  - Execution ("Mutation") Time
  - Average Power Usage
- **Energy** =  $\beta_0 + \beta_1 \text{Prod} + \beta_2 \text{GC} + \beta_3 \text{Mut} + \beta_4 \text{Power}$

# Application-Specific Energy Model

| Program                        | Parfib     | Sumeuler  | Nbody      | DFT    | Queens | MatMult | Minimax |
|--------------------------------|------------|-----------|------------|--------|--------|---------|---------|
| Coefficient                    |            |           |            |        |        |         |         |
| $\hat{\beta}_0$ (Intercept)    | -1.953e+03 | 333.2     | -3.966e+02 | 567.54 | -287.7 | -287.7  | 152.59  |
| $\hat{\beta}_1$ (productivity) | -4.944e-01 | -0.30394  | -1.232e-01 | 0.111  | -0.347 | -0.347  | 570.06  |
| $\hat{\beta}_2$ (Mutation)     | -1.983e+03 | 13.57     | -3.128e+02 | 448.08 | -85.84 | -85.8   | 666.77  |
| $\hat{\beta}_2$ (GC)           | 285.09     | 7.378e+03 | -7.251e+02 | 552.3  | 11.43  | 11.43   | 0.094   |
| $\hat{\beta}_3$ (power)        | 4.749e+01  | 11.84     | 5.016      | -10.14 | 11.08  | -0.34   | -1.833  |

Table 3.  $\hat{\beta}$  coefficients for the complete dataset

High significance in almost all cases ( $p < 0.01$ ) for Productivity and Power  
Mutation and GC are also significant in many cases ( $p < 0.05$ )  
Only minimax showed low significance ( $p > 0.1$ )



# How well do the Models fit the Results?

| Parfib                    |             | Sumeuler                  |             | NBody                     |             | DFT                       |             | Queens                    |             | Matmult                   |             | Minimax                   |             |
|---------------------------|-------------|---------------------------|-------------|---------------------------|-------------|---------------------------|-------------|---------------------------|-------------|---------------------------|-------------|---------------------------|-------------|
| <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> | <i>Adj. R<sup>2</sup></i> | <i>RMSE</i> |
| 97.6%                     | 186.75      | 98.7%                     | 243.41      | 98.8%                     | 653.26      | 98.1%                     | 251.96      | 98.5%                     | 281.48      | 97.4%                     | 189.65      | 99.2%                     | 349.71      |

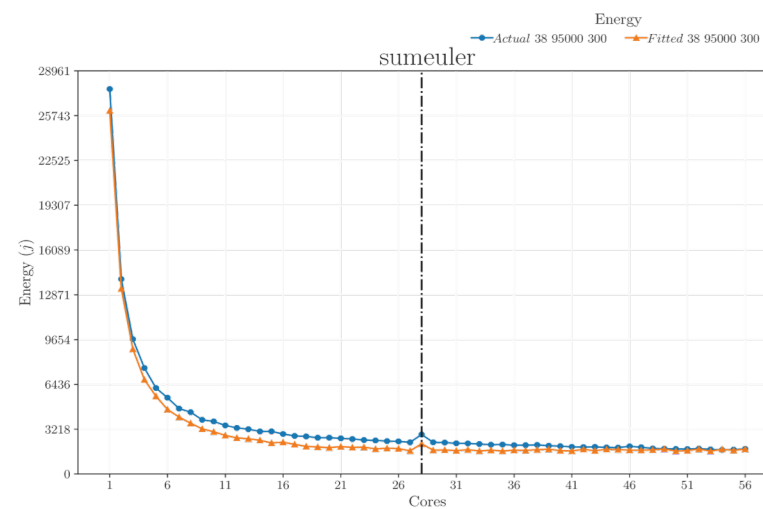
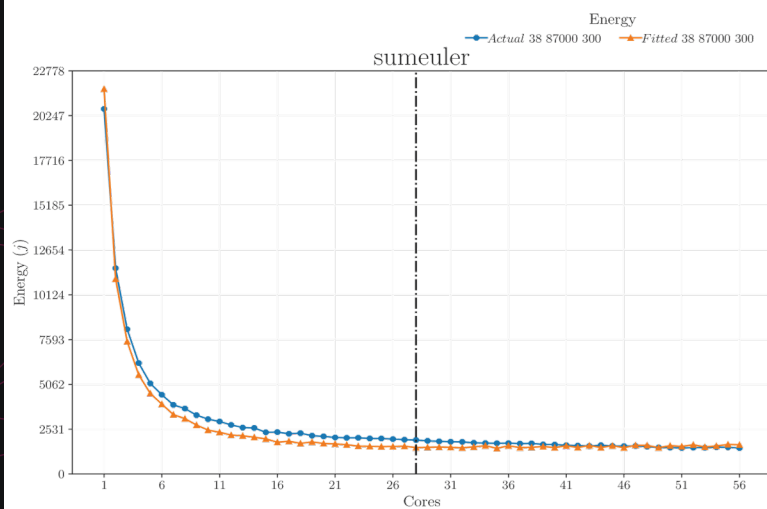
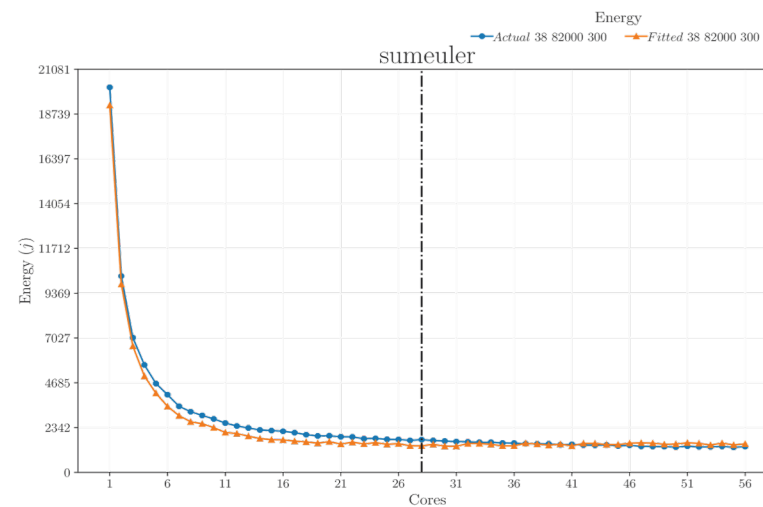
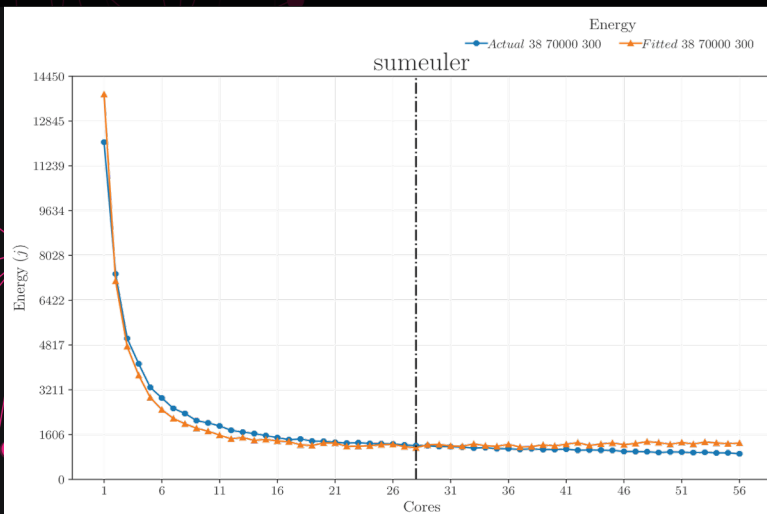
**Table 1.** *Adjusted R<sup>2</sup>* and root mean square error (*RMSE*) values for the complete dataset

**R<sup>2</sup> values show good fits between models and predictions in all cases (> 97%)**

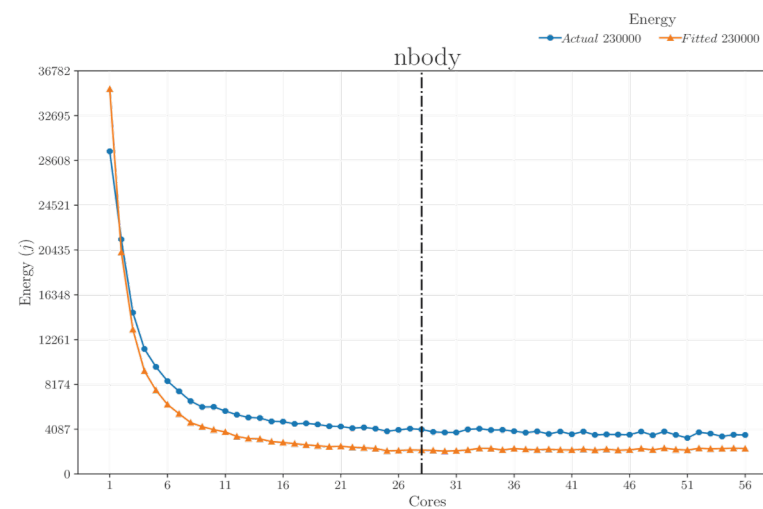
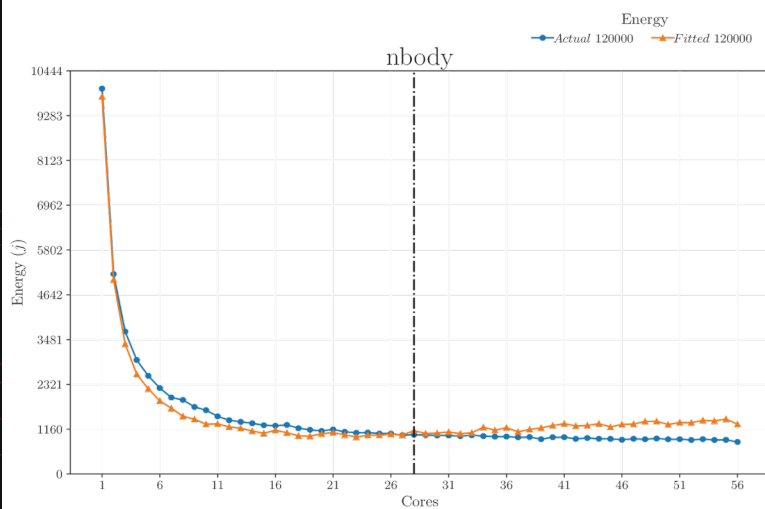
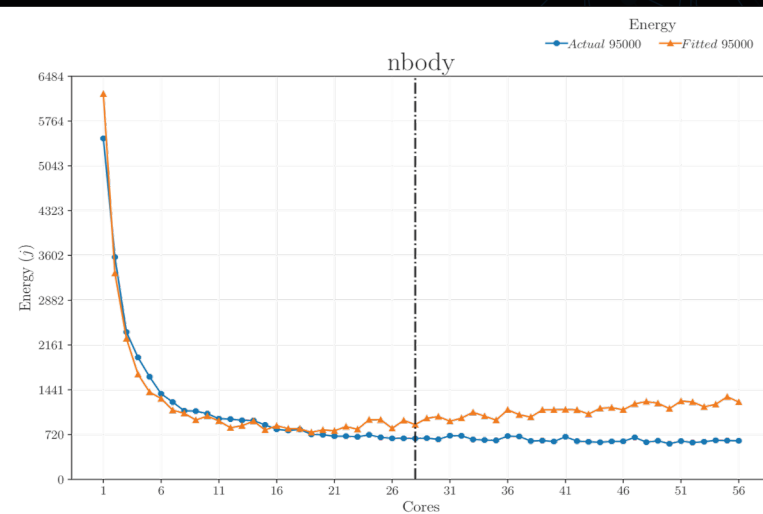
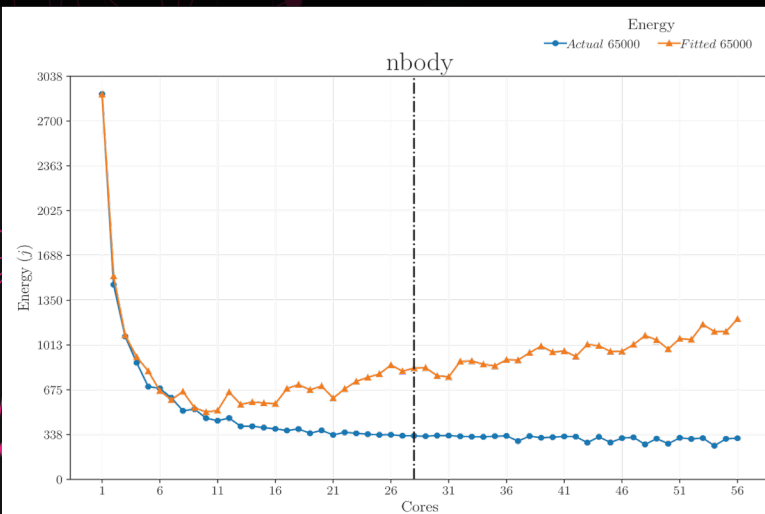
## Experimental Results for Application-Specific Models

- 4 new problem sizes compared with predictions from models
  - Two smaller and two larger data sets considered

# Application-specific model: Euler Summation

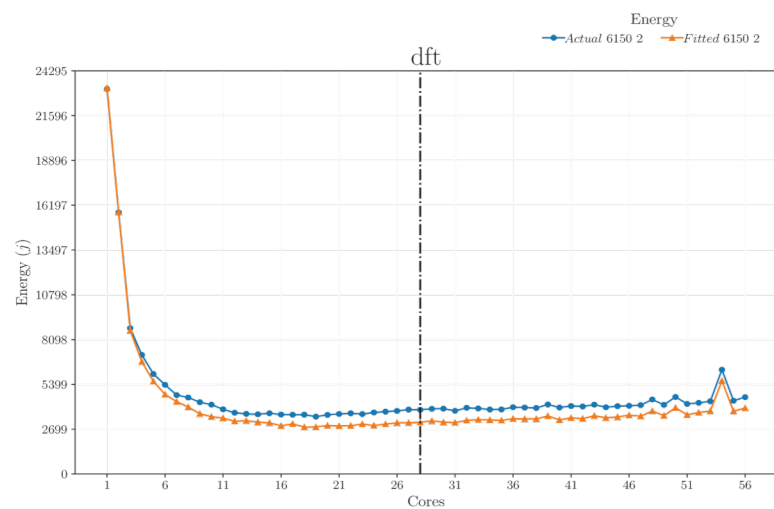
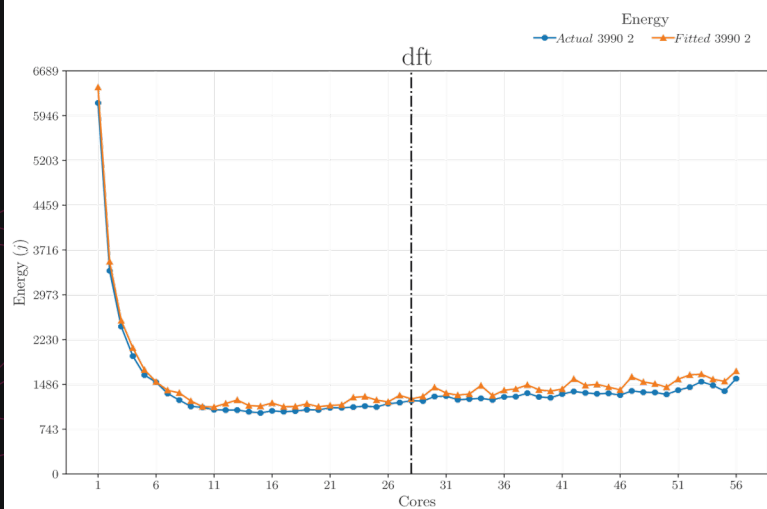
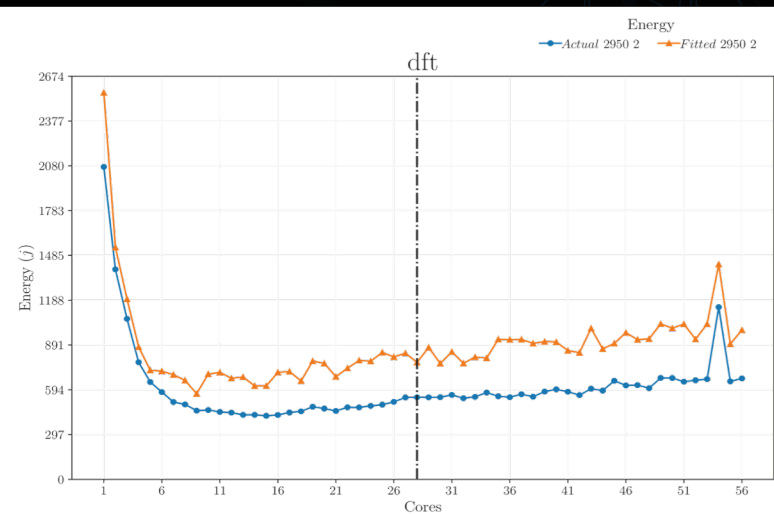
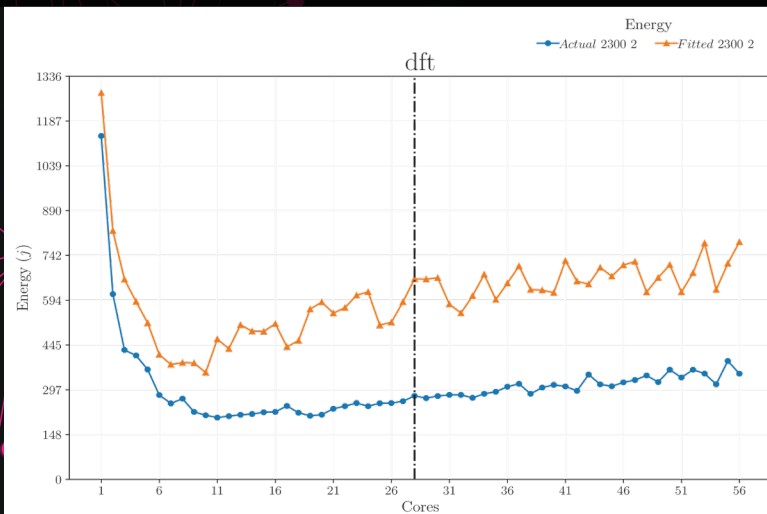


# Application-specific model: N-Body

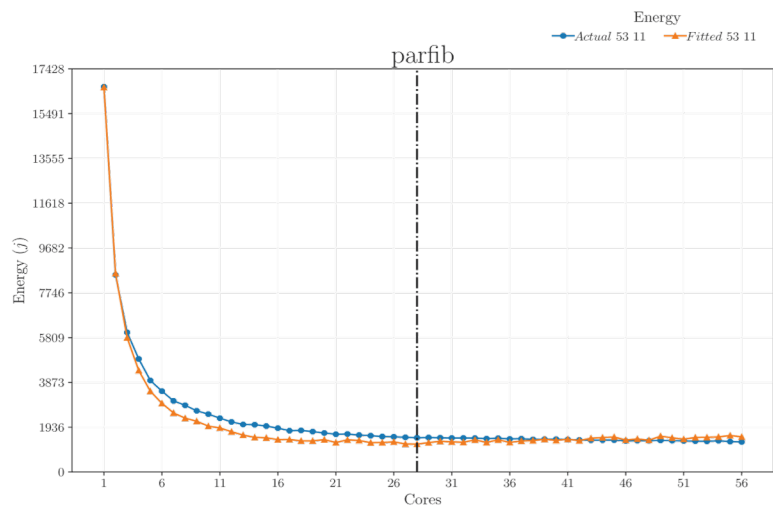
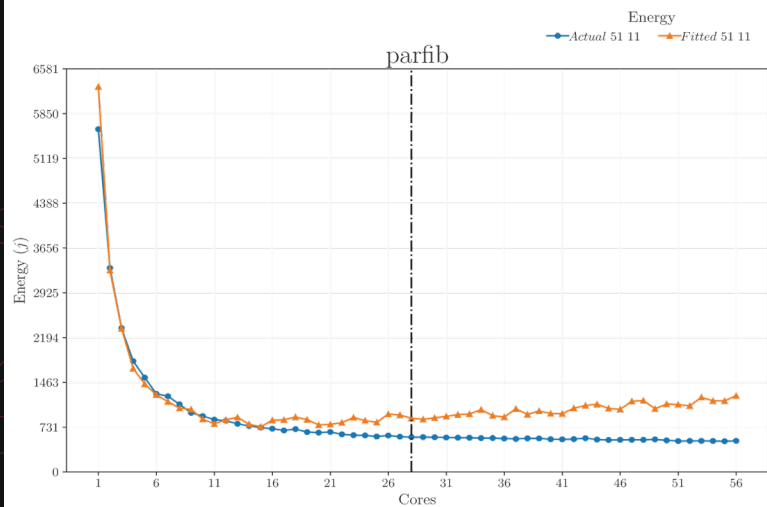
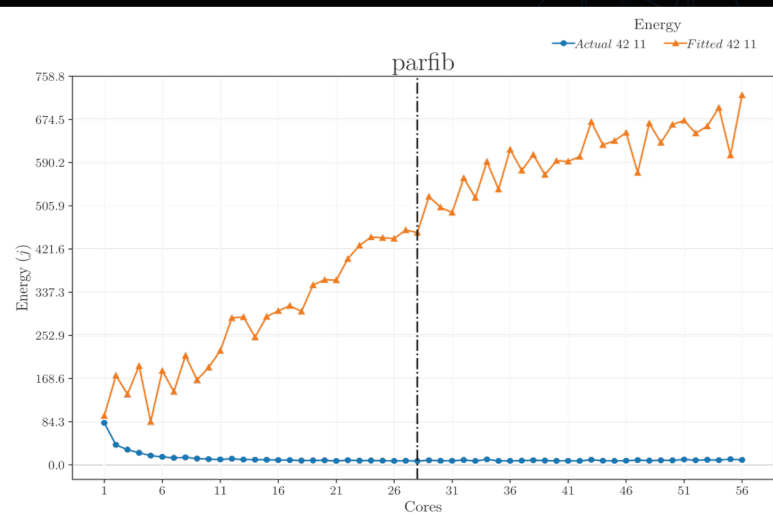
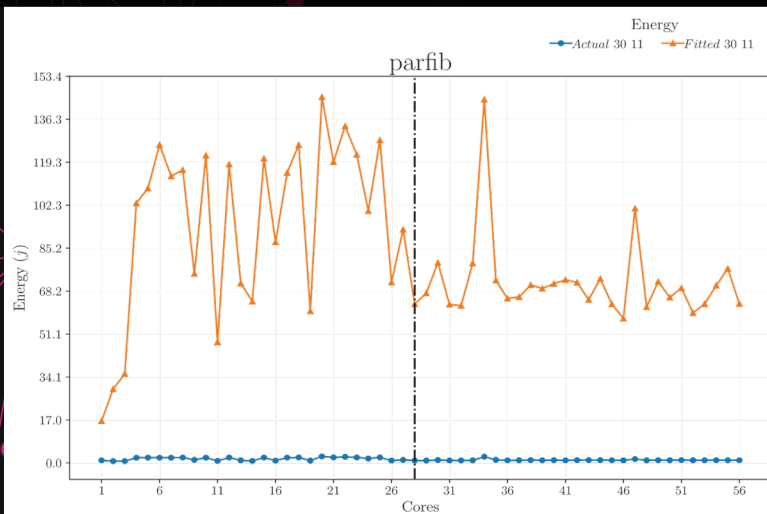




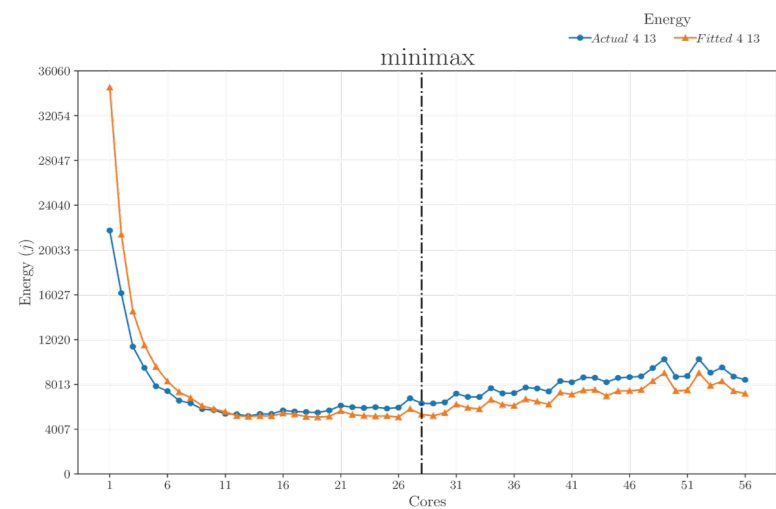
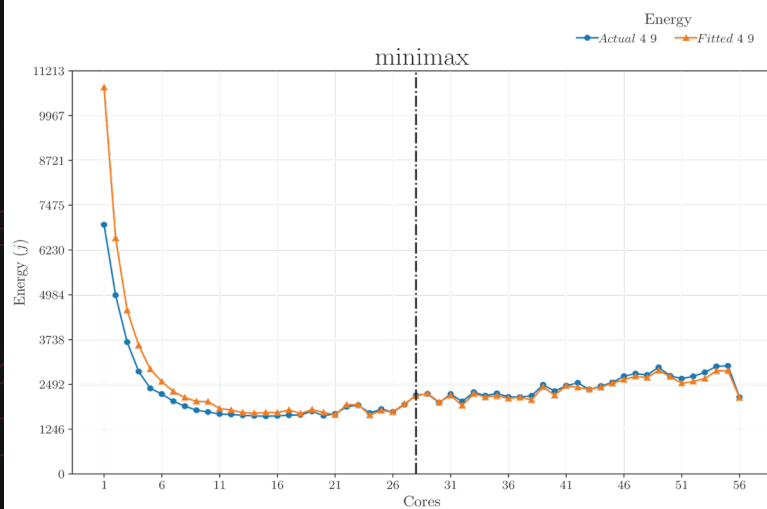
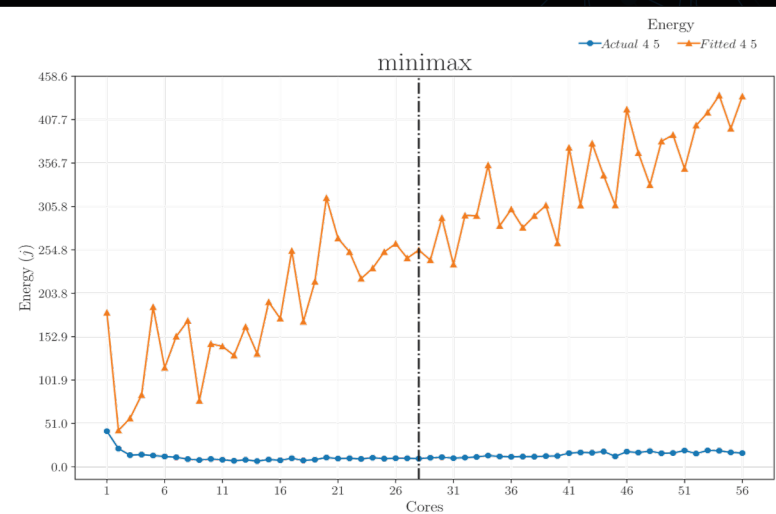
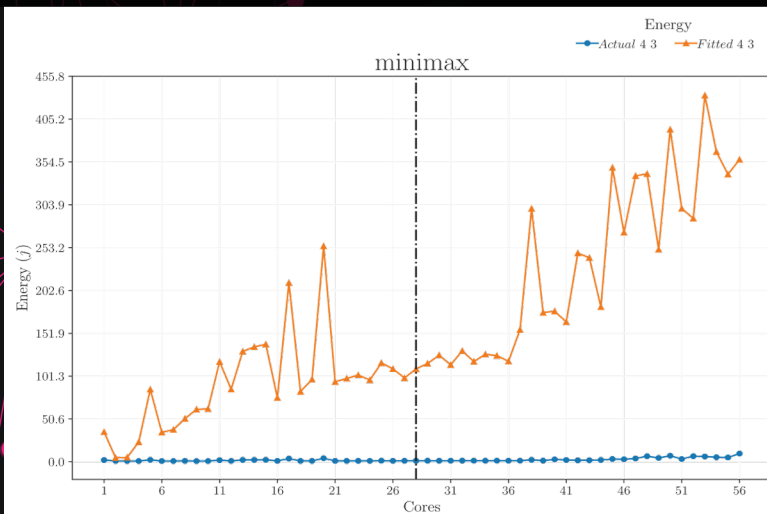
# Application-specific model: DFT



# Application-specific model: Parallel Fibonacci



# Application-specific model: Minimax



## Discussion

- **Application-specific Models are a Good fit at larger sizes**
  - Even to previously unseen data sets
- **There is often a Poor Fit at smaller sizes**
  - More likely to be affected by small variations?
  - Less consistent energy usage?



## New Research Question

**Q3: Can we construct a “model of models”?**



# A Generalised Energy Model

Residuals:

| Min      | 1Q     | Median | 3Q    | Max    |
|----------|--------|--------|-------|--------|
| -12291.8 | -400.0 | -167.2 | 291.7 | 7129.2 |

Coefficients:

|              | Estimate   | Std. Error | t value | Pr(> t ) |     |
|--------------|------------|------------|---------|----------|-----|
| (Intercept)  | -188.8143  | 28.3854    | -6.652  | 2.99e-11 | *** |
| power        | 4.9193     | 0.5146     | 9.559   | < 2e-16  | *** |
| mutation     | -1580.8171 | 67.9912    | -23.250 | < 2e-16  | *** |
| gc           | -1544.8994 | 68.0120    | -22.715 | < 2e-16  | *** |
| productivity | 0.1210     | 0.0086     | 14.071  | < 2e-16  | *** |
| total time   | 1615.2182  | 67.9903    | 23.757  | < 2e-16  | *** |

---

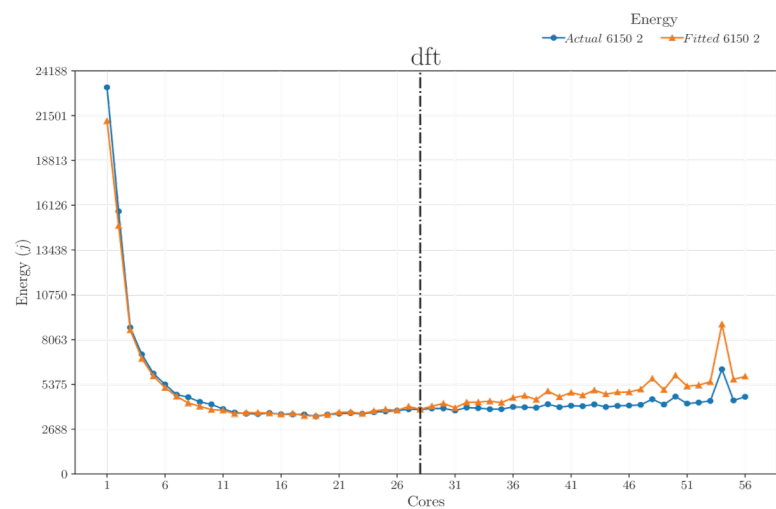
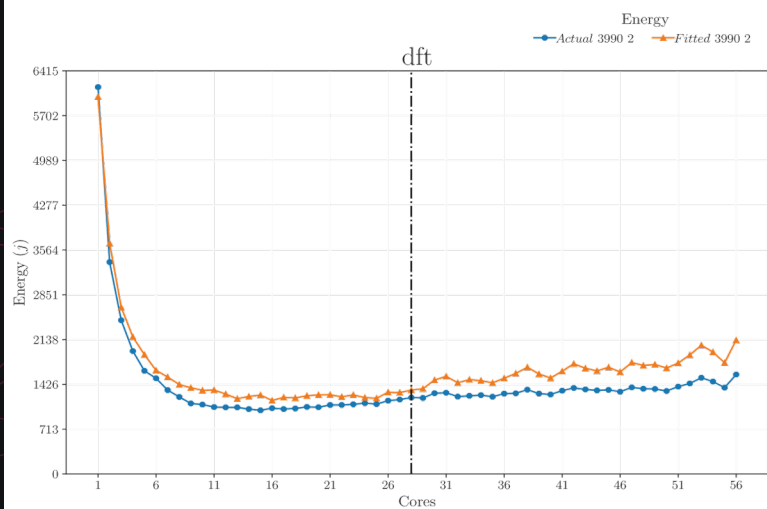
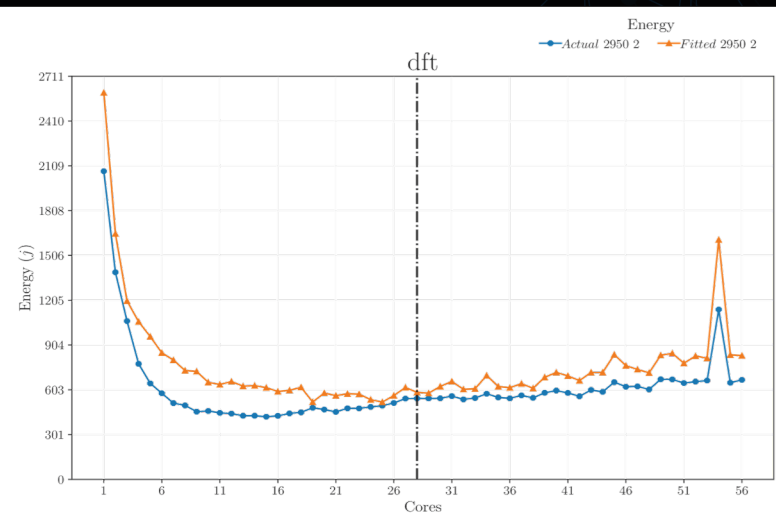
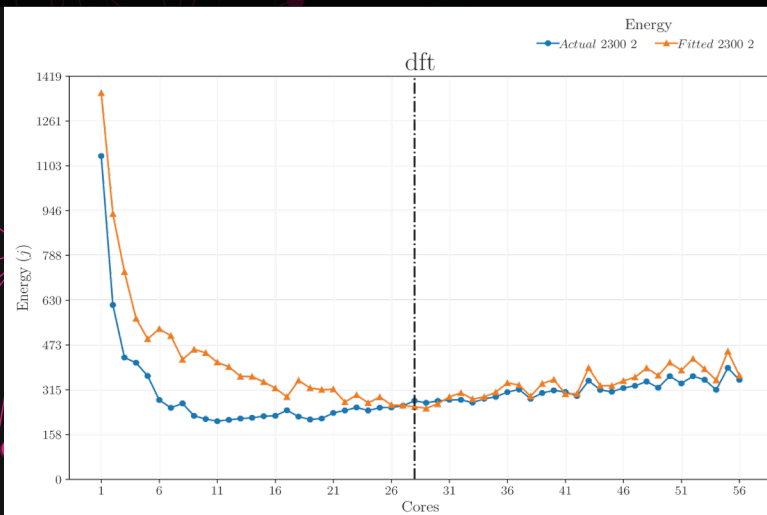
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 819 on 15681 degrees of freedom

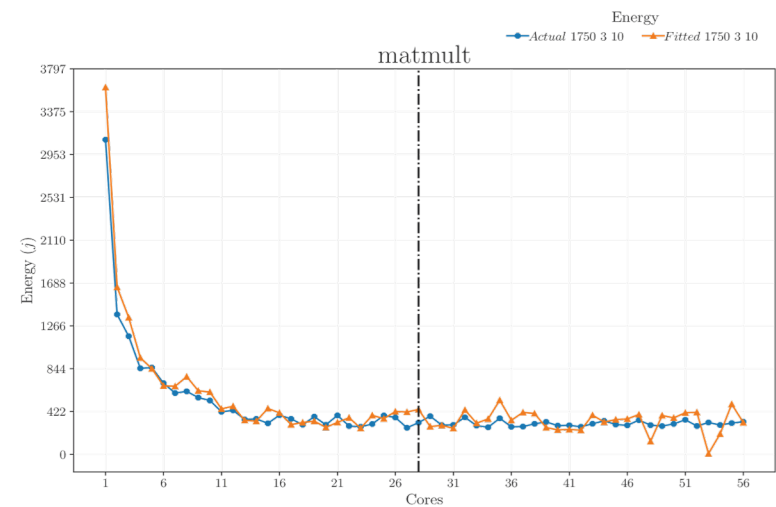
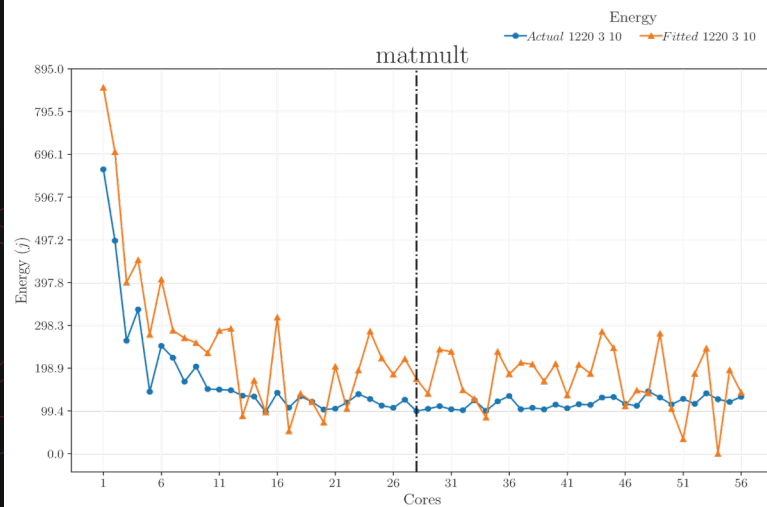
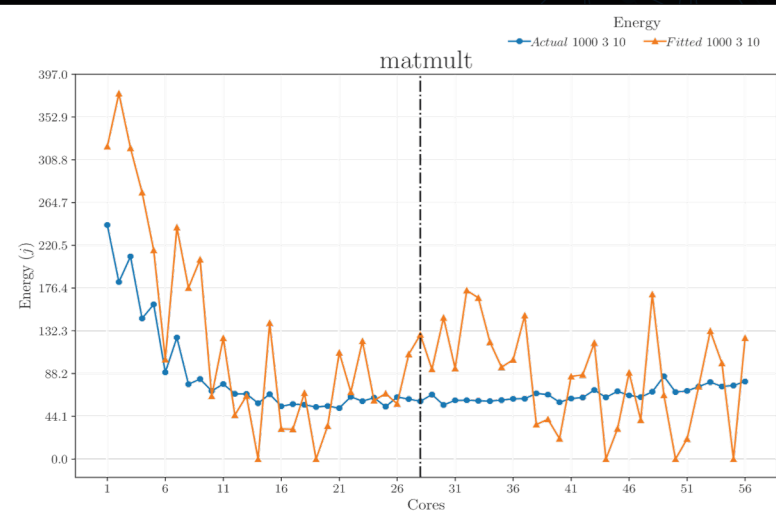
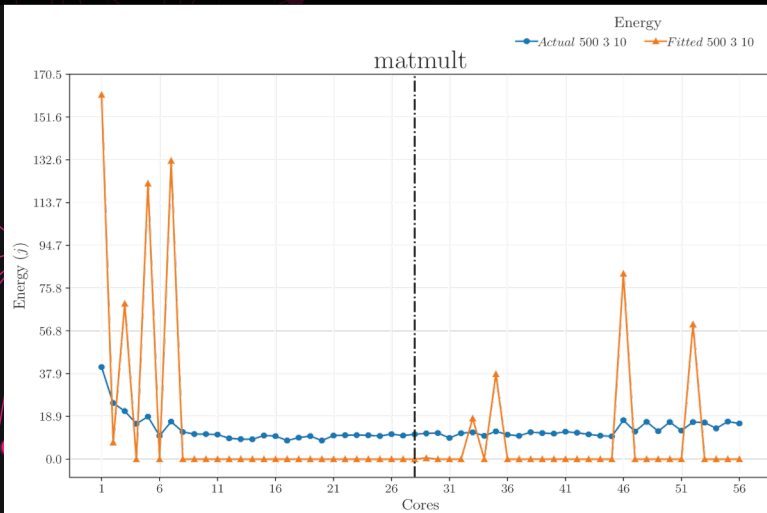
Multiple R-squared: 0.9398, Adjusted R-squared: 0.9397

F-statistic: 4.892e+04 on 5 and 15681 DF, p-value: < 2.2e-16

# General model: DFT

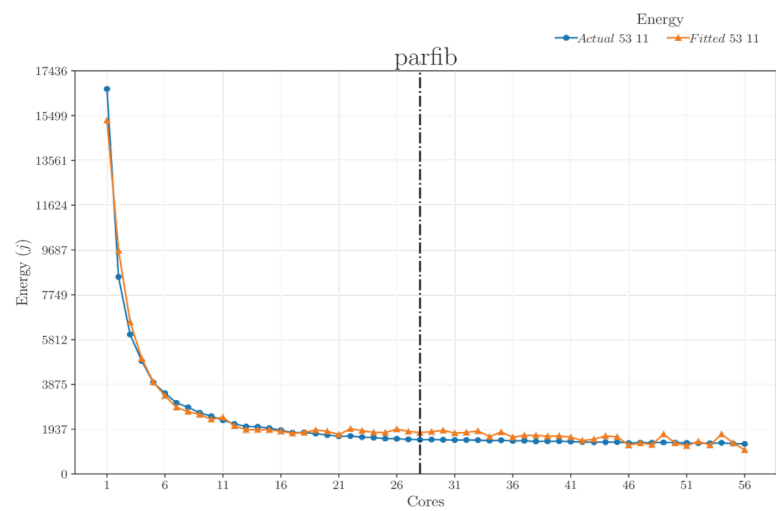
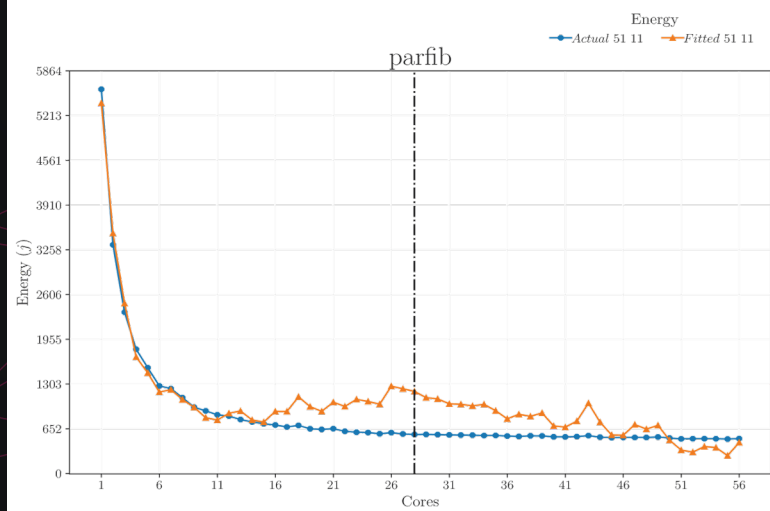
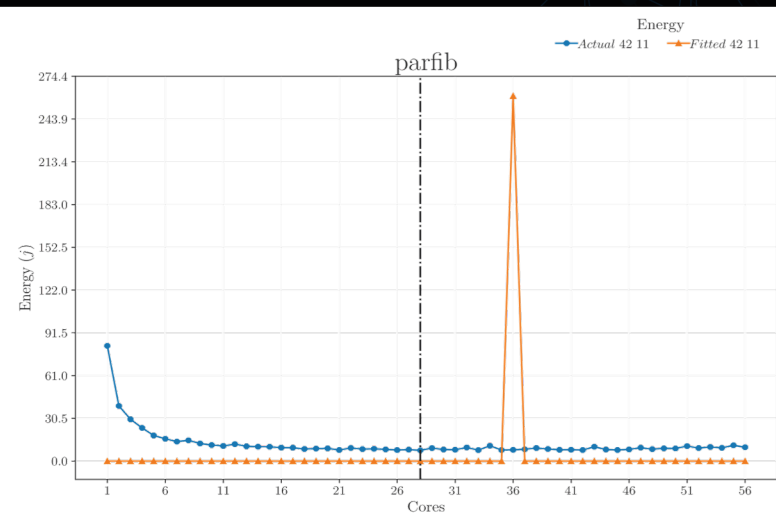
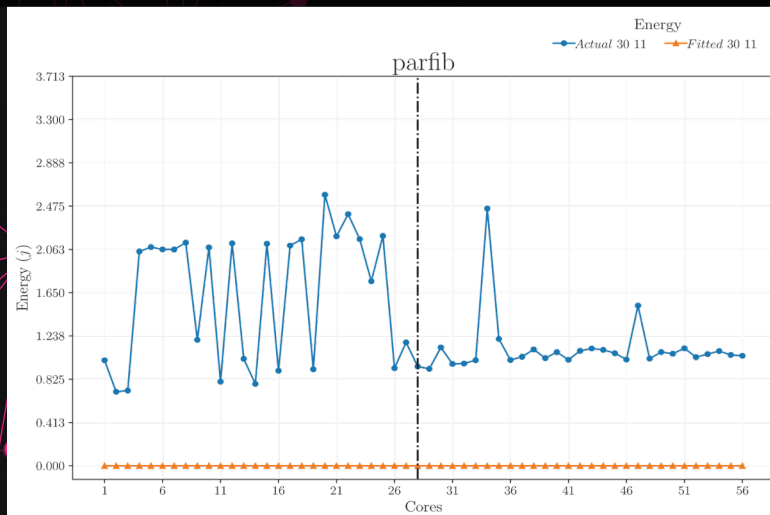


# General model: Matrix Multiplication

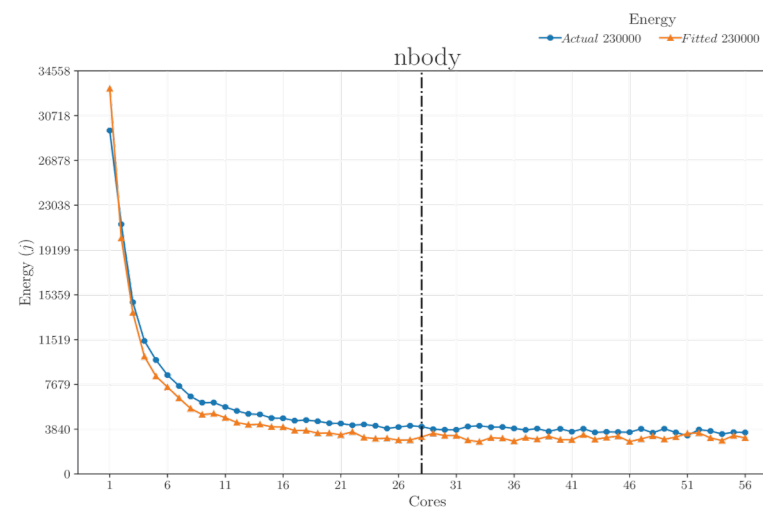
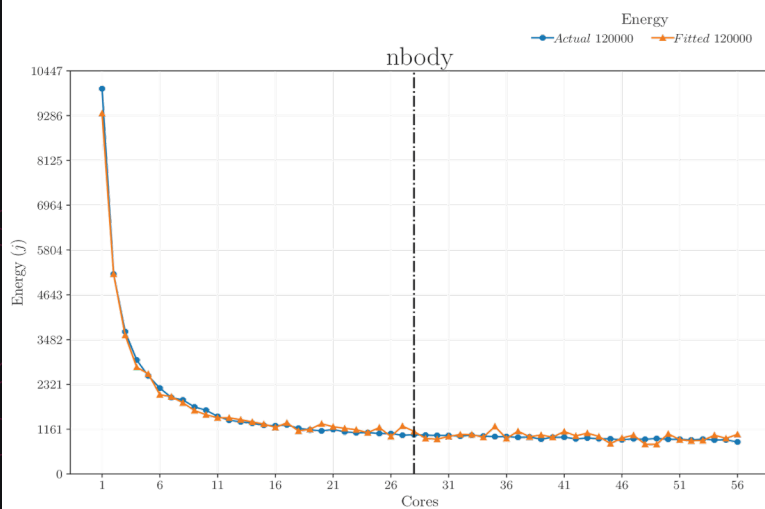
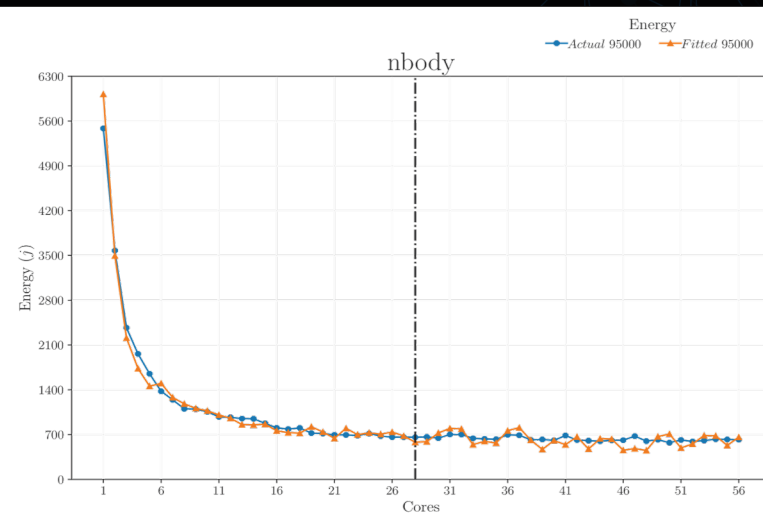
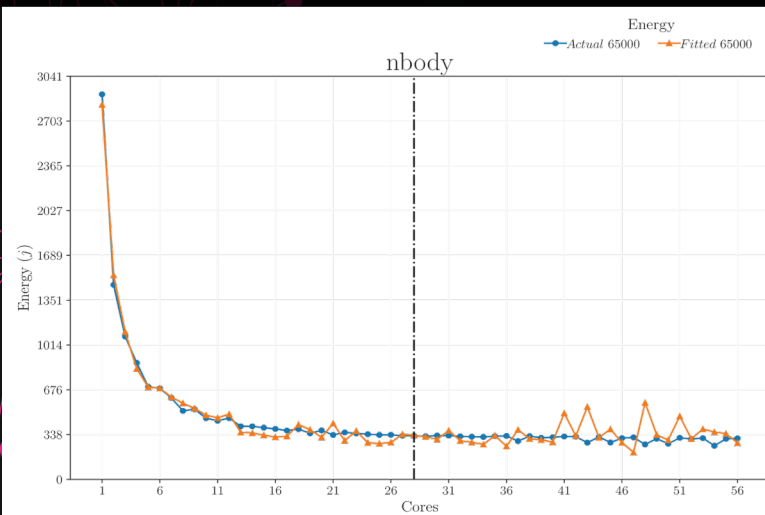




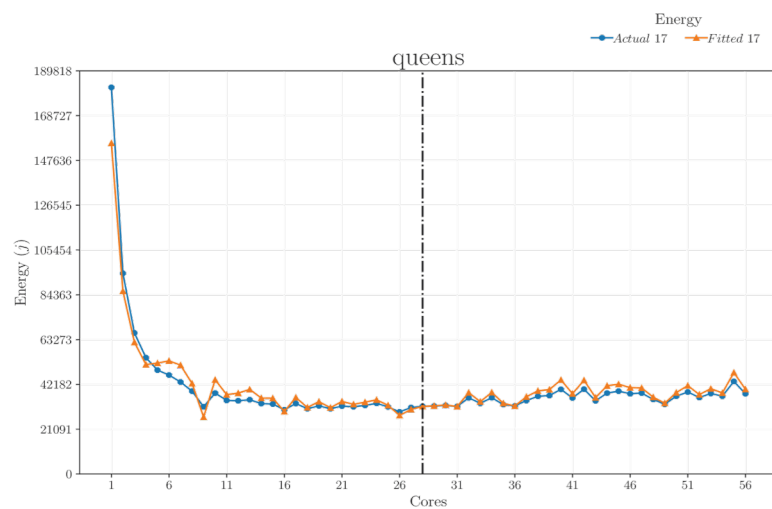
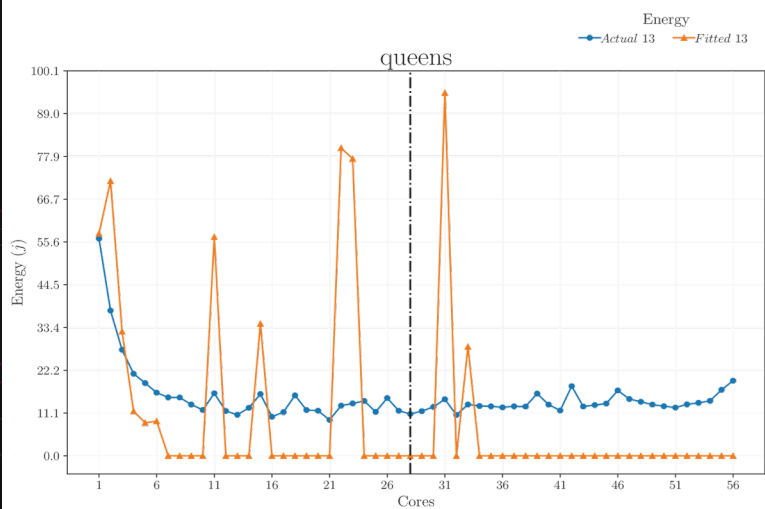
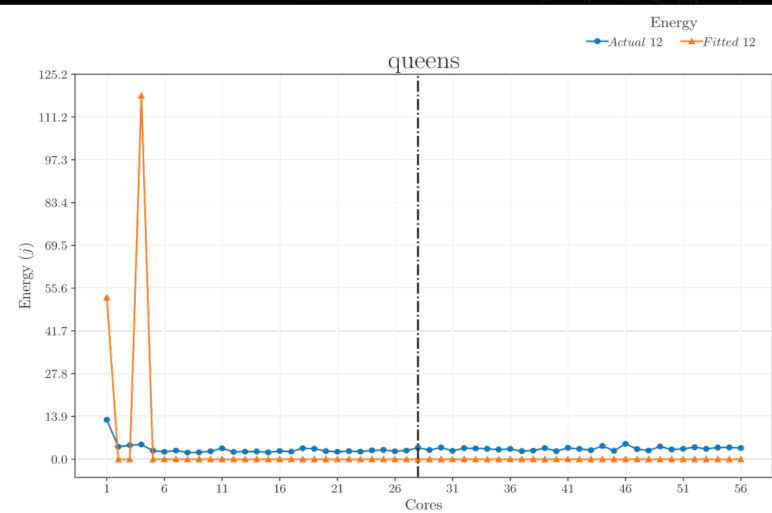
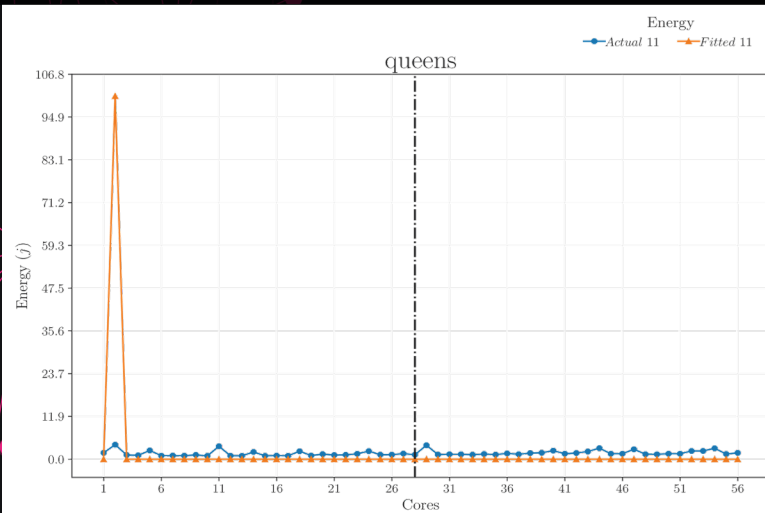
# General model: Parallel Fibonacci



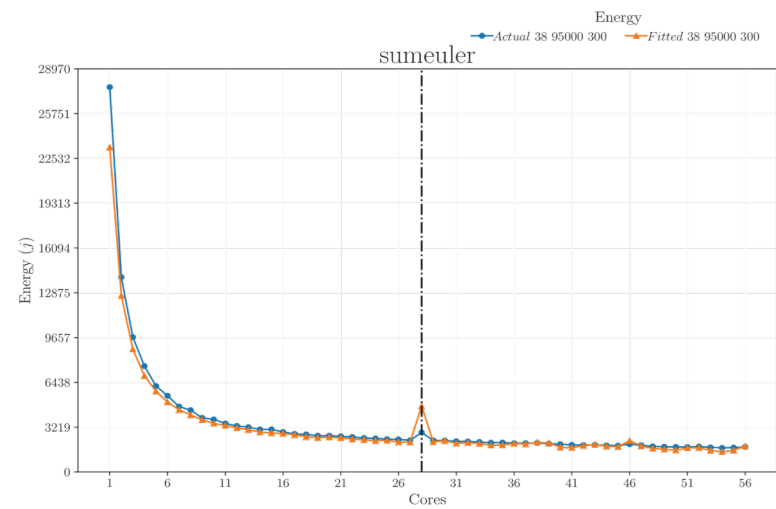
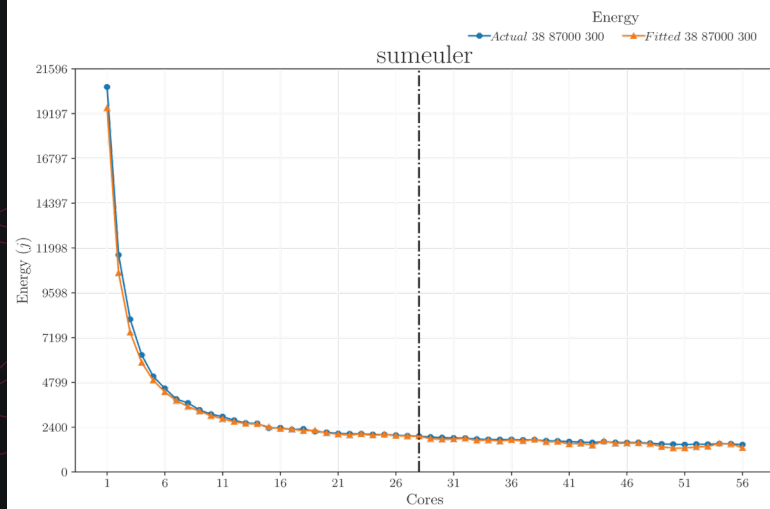
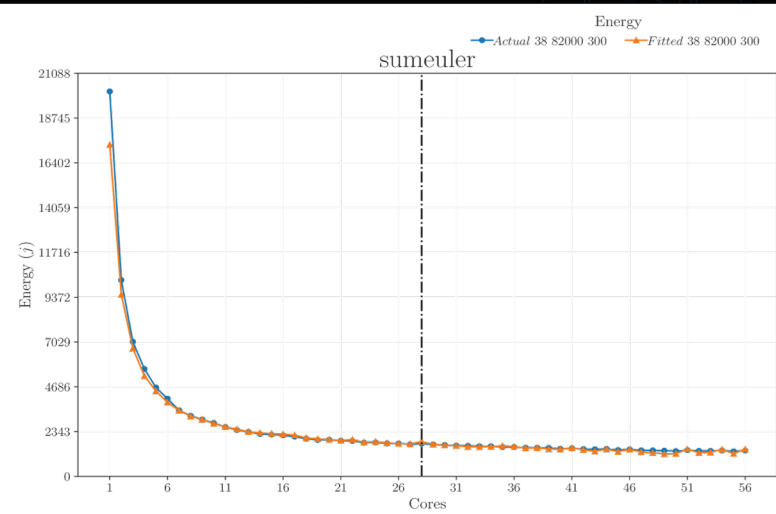
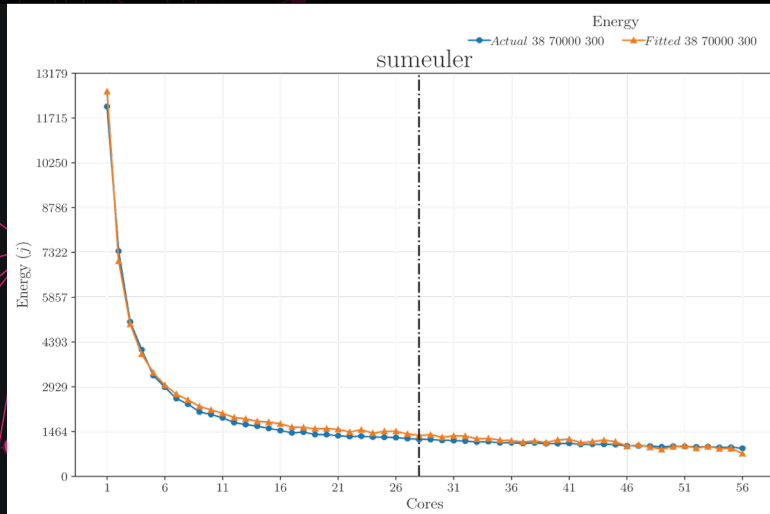
# General model: N-Body



# General model: N-Queens

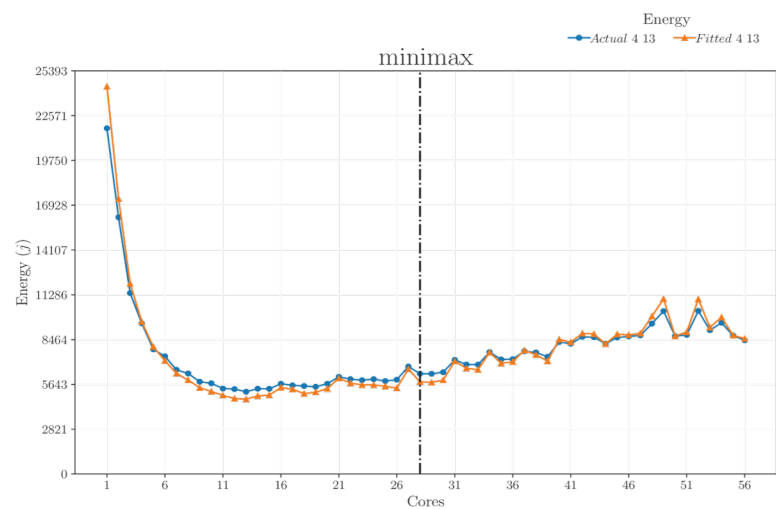
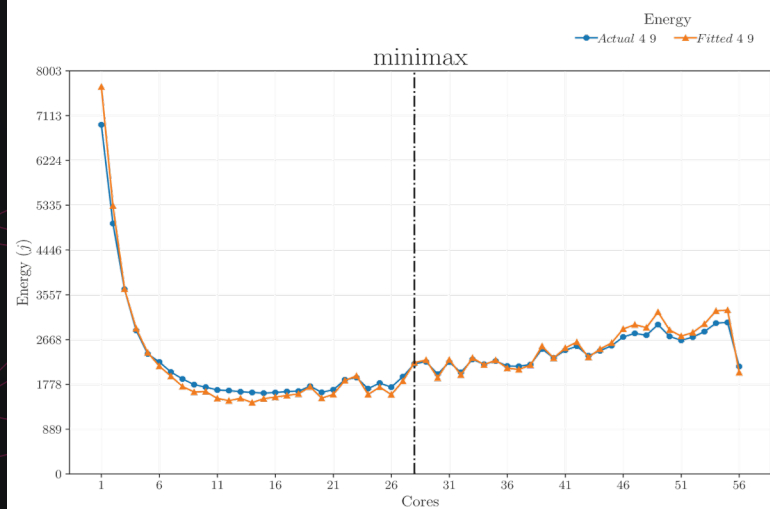
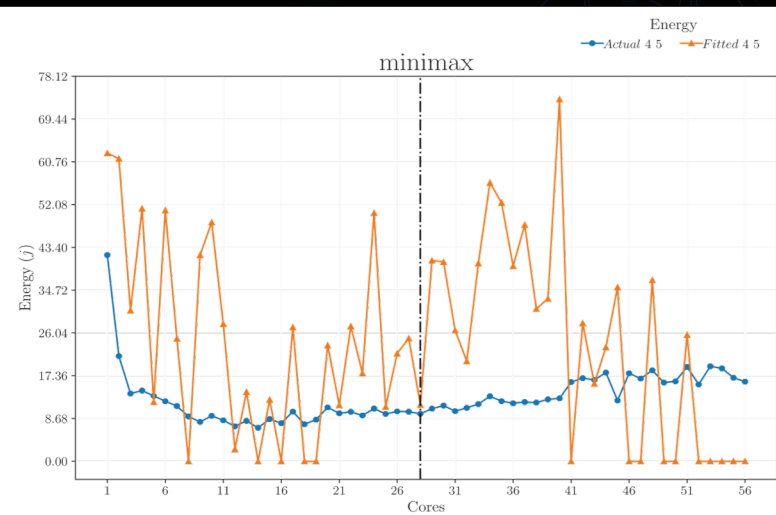
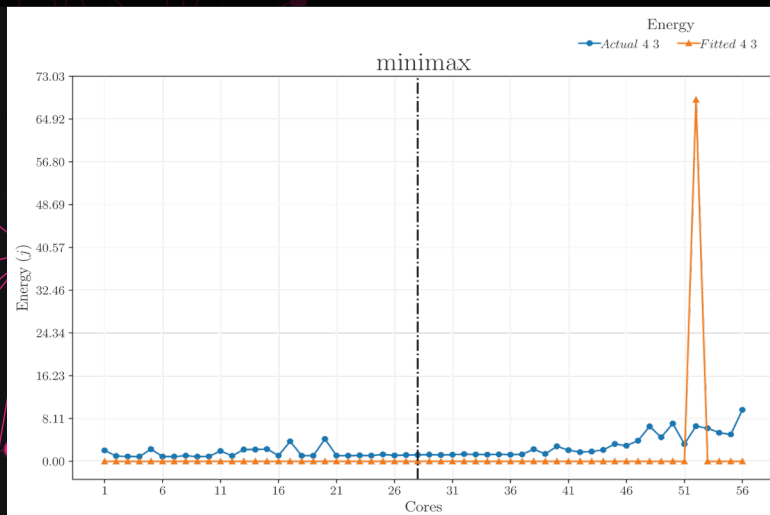


# General model: Euler Summation





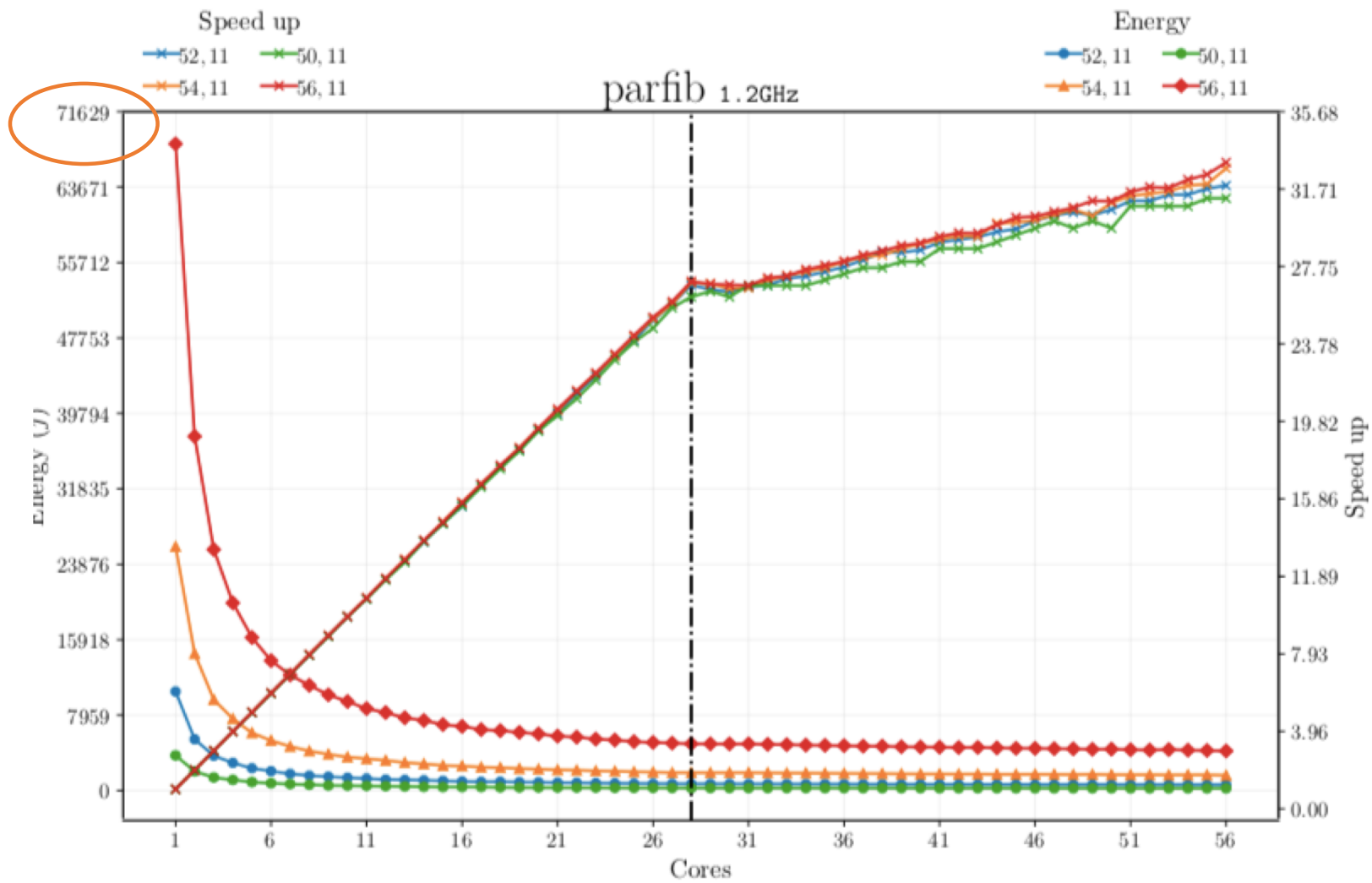
# General model: Minimax

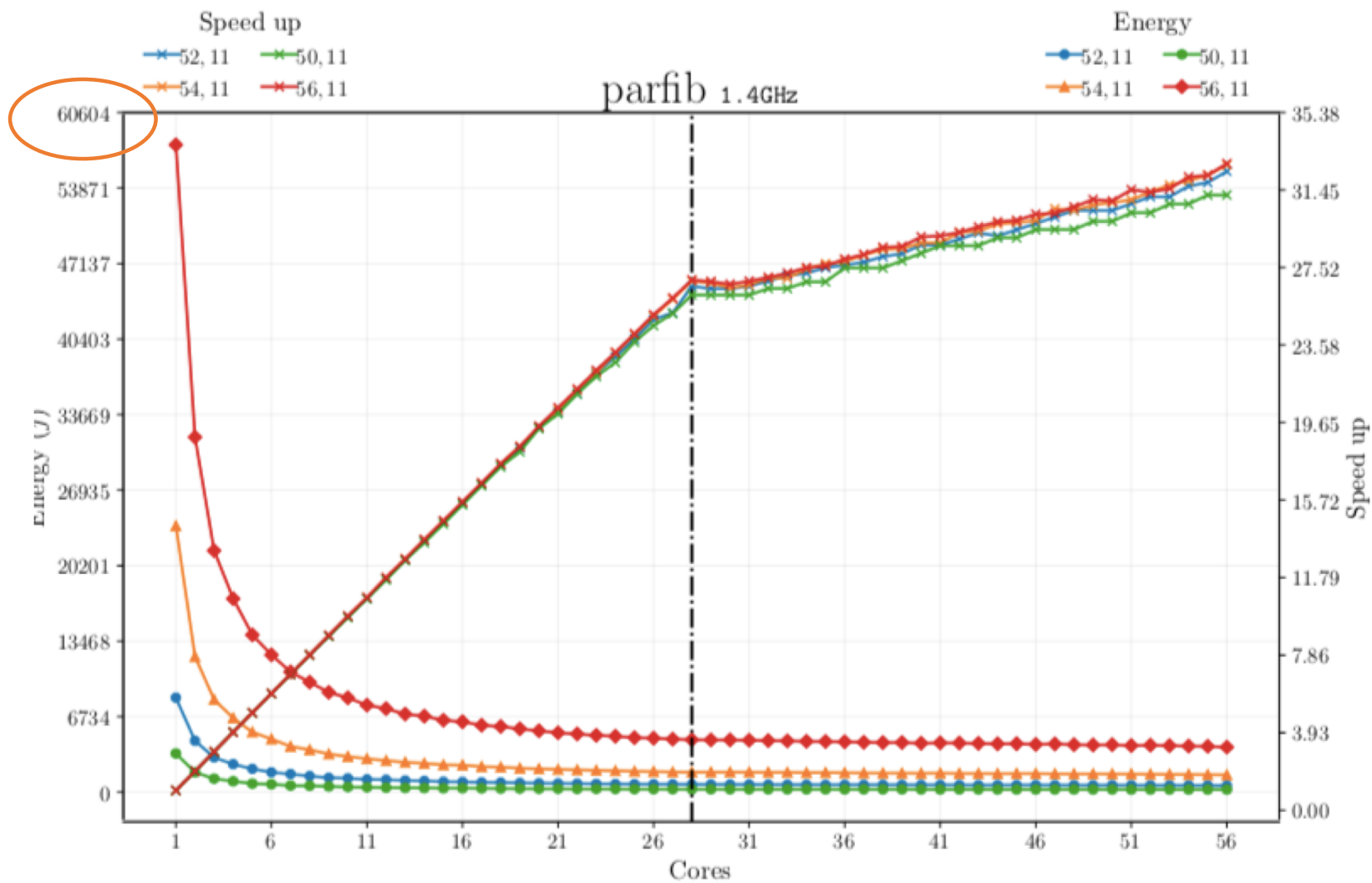


## New Research Question

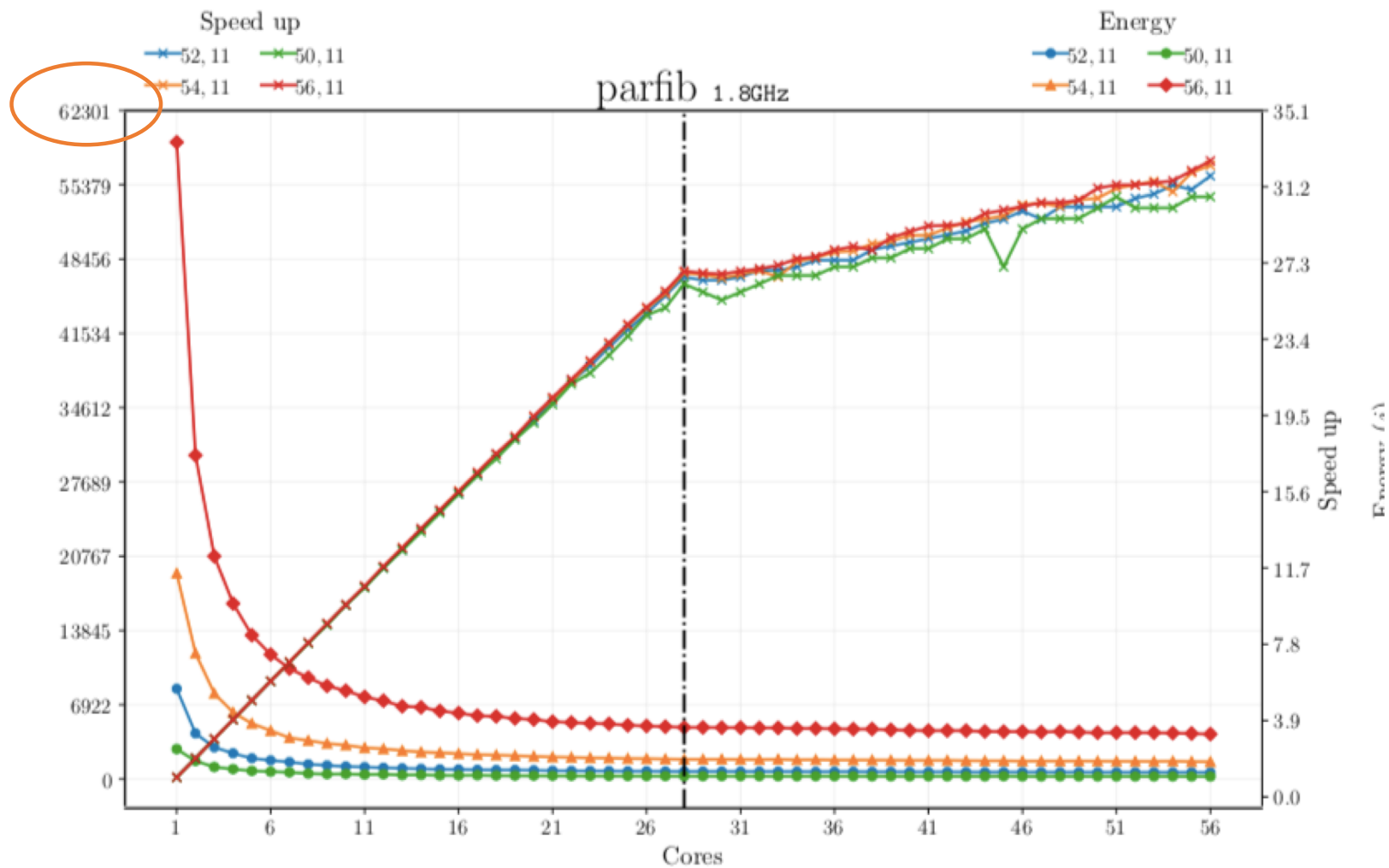
**Q4: How does clock frequency affect energy usage?**

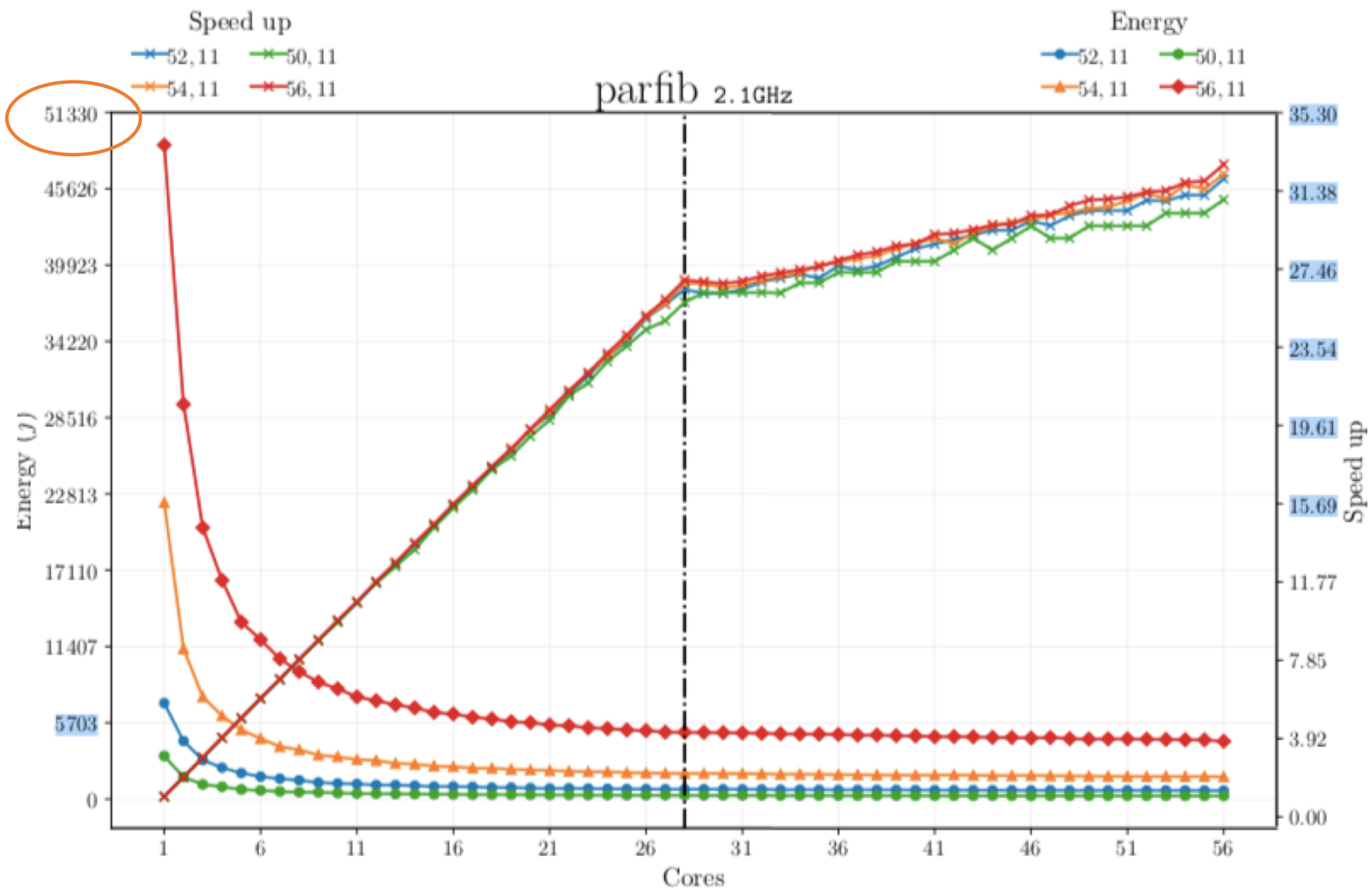


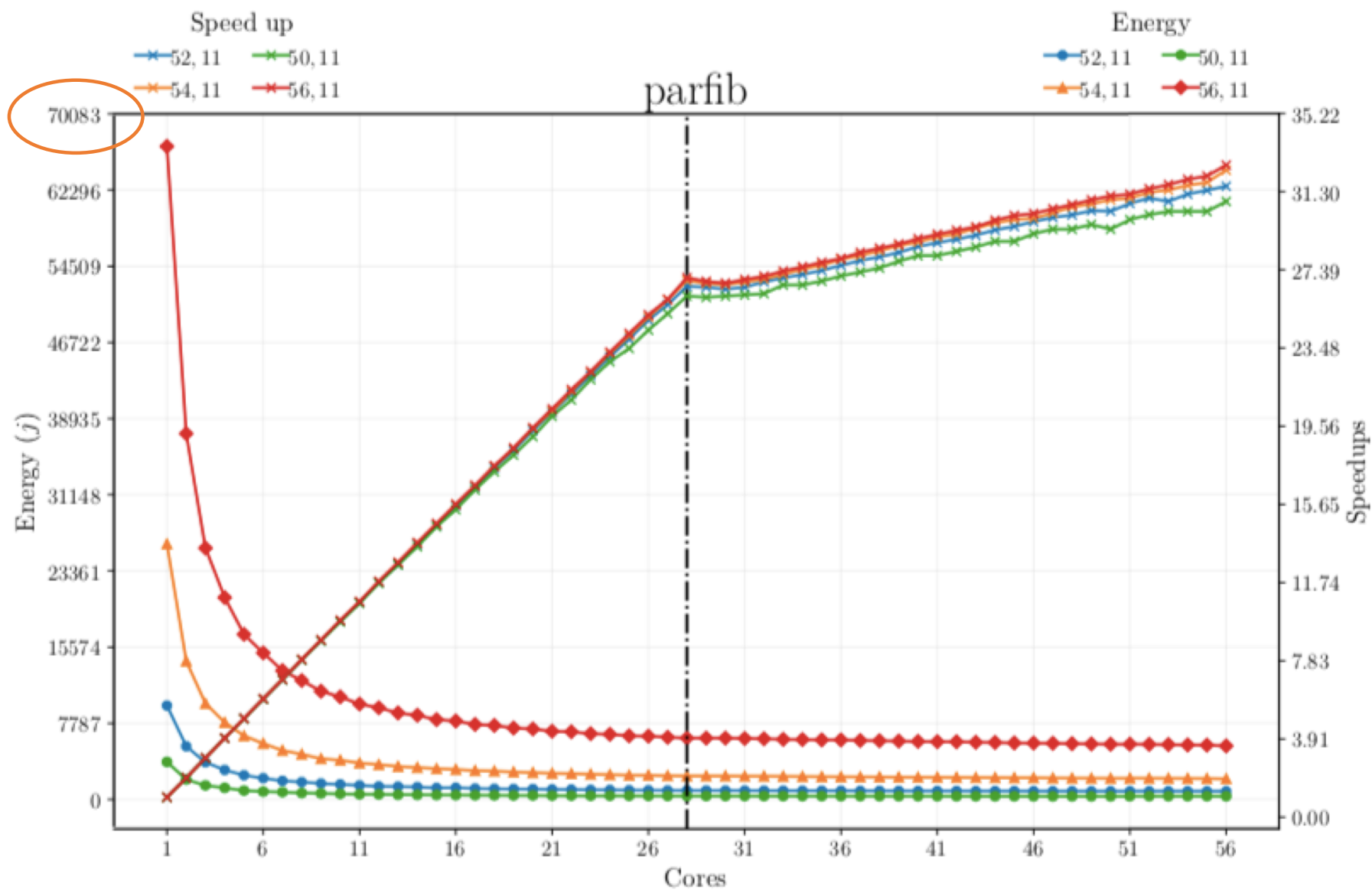












## Discussion: Energy Usage v Clock Speed

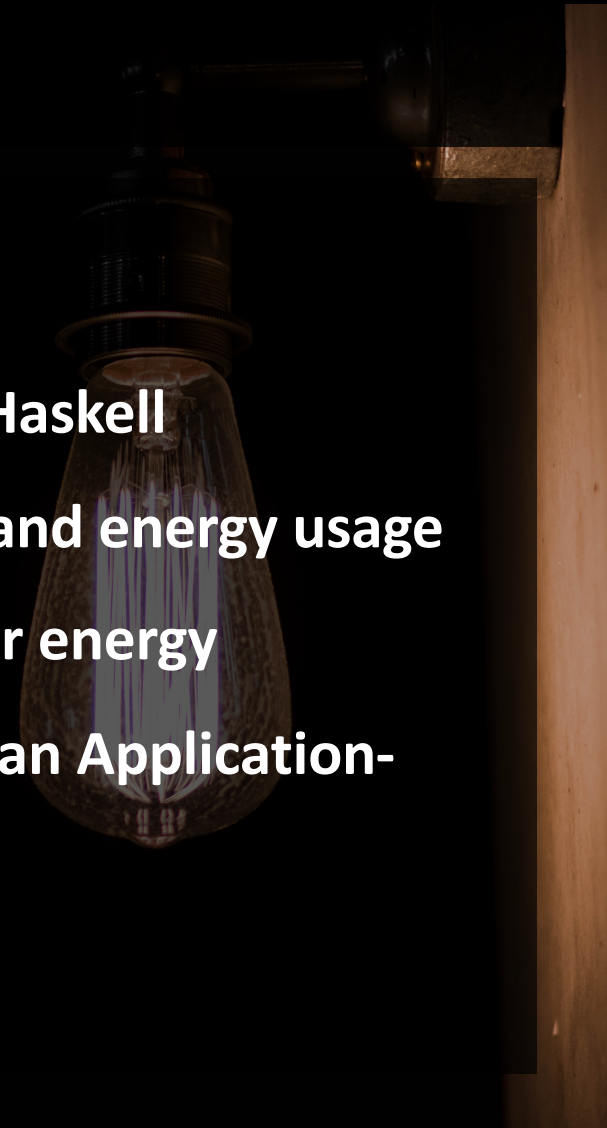
- Maximum clock frequency did not yield best energy usage
- Least energy usage at 2.1GHz





## Conclusions

- Runtime energy usage is predictable for Parallel Haskell
- General close correlation between performance and energy usage
- BUT highest clock frequency does not equal lower energy
- General model gives better overall predictions than Application-Specific ones



# Thank you!

- [kevin@kevinhammond.net](mailto:kevin@kevinhammond.net)
- [ya8@st-andrews.ac.uk](mailto:ya8@st-andrews.ac.uk)

