Topics in Performance Evaluation

Dror Feitelson
Hebrew University

Experimental Design and Analysis of Variation

The questions:

1) What system configurations to simulate

2) What do the measurement results mean

Factorial Design

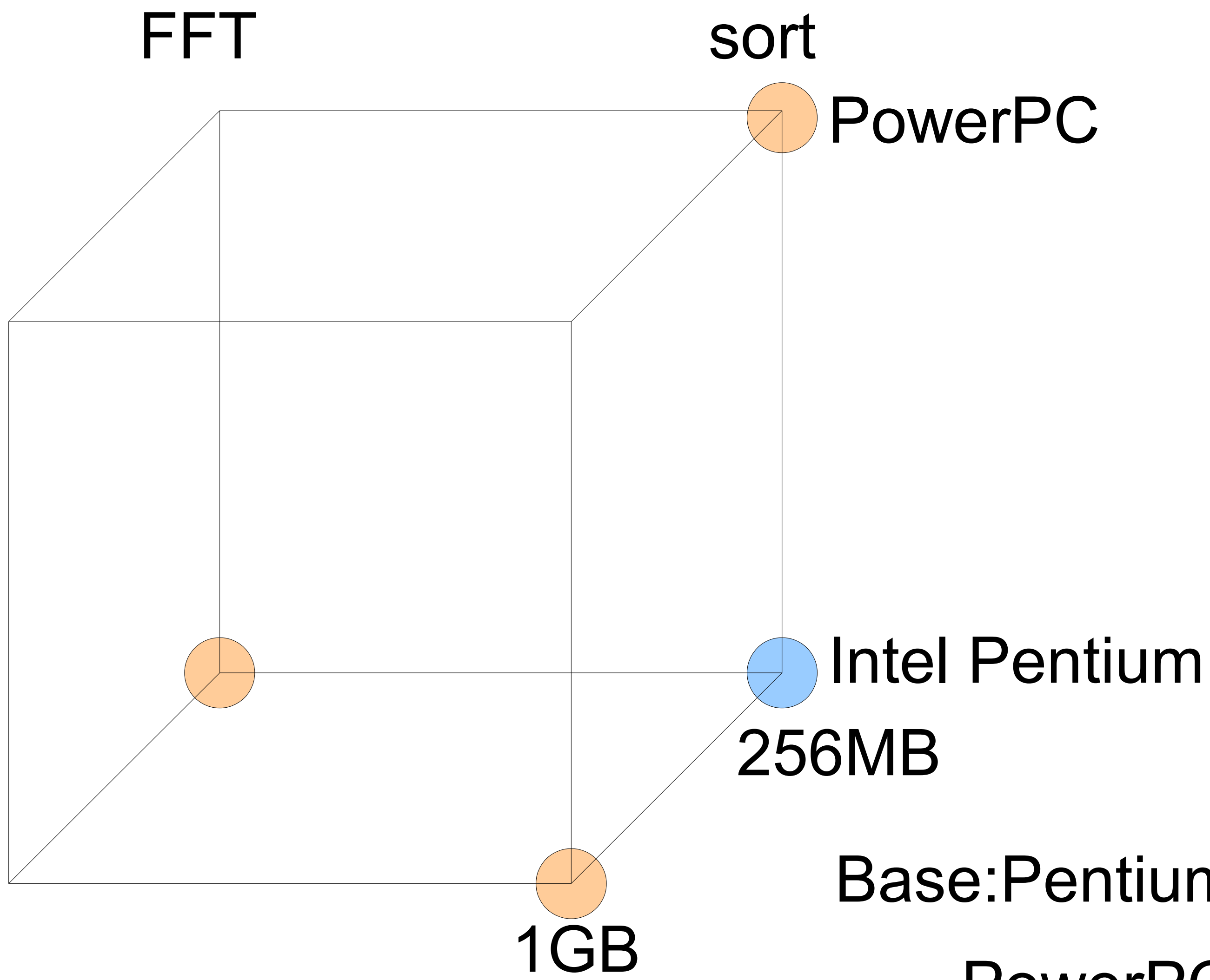
- A Factor – something that affects performance
  - The model of the CPU
  - The amount of memory you have
  - Which benchmark is being measured
- A level – one of the values assumed by a factor
  - Pentium Pro, Pentium III, or Pentium IV
  - 256MB, 512MB, or 1GB
  - Sorting, FFT, compilation, copying a file
- A design – setting the number of experiments, and which combination of levels will be used in each one

Simple factorial design

- Select a base configuration and measure it
- For each factor independently, set the different levels and perform measurements
- With k factors and n_i levels, the number of experiments is

$$1+\sum_{i}^{k}(n_i-1)$$

- Problem: does not identify interactions among the factors
  - Example: different benchmarks may have different sensitivity to memory size



Base:Pentium + 256MB + sort

PowerPC + 256MB + sort

Pentium + 256MB + FFT

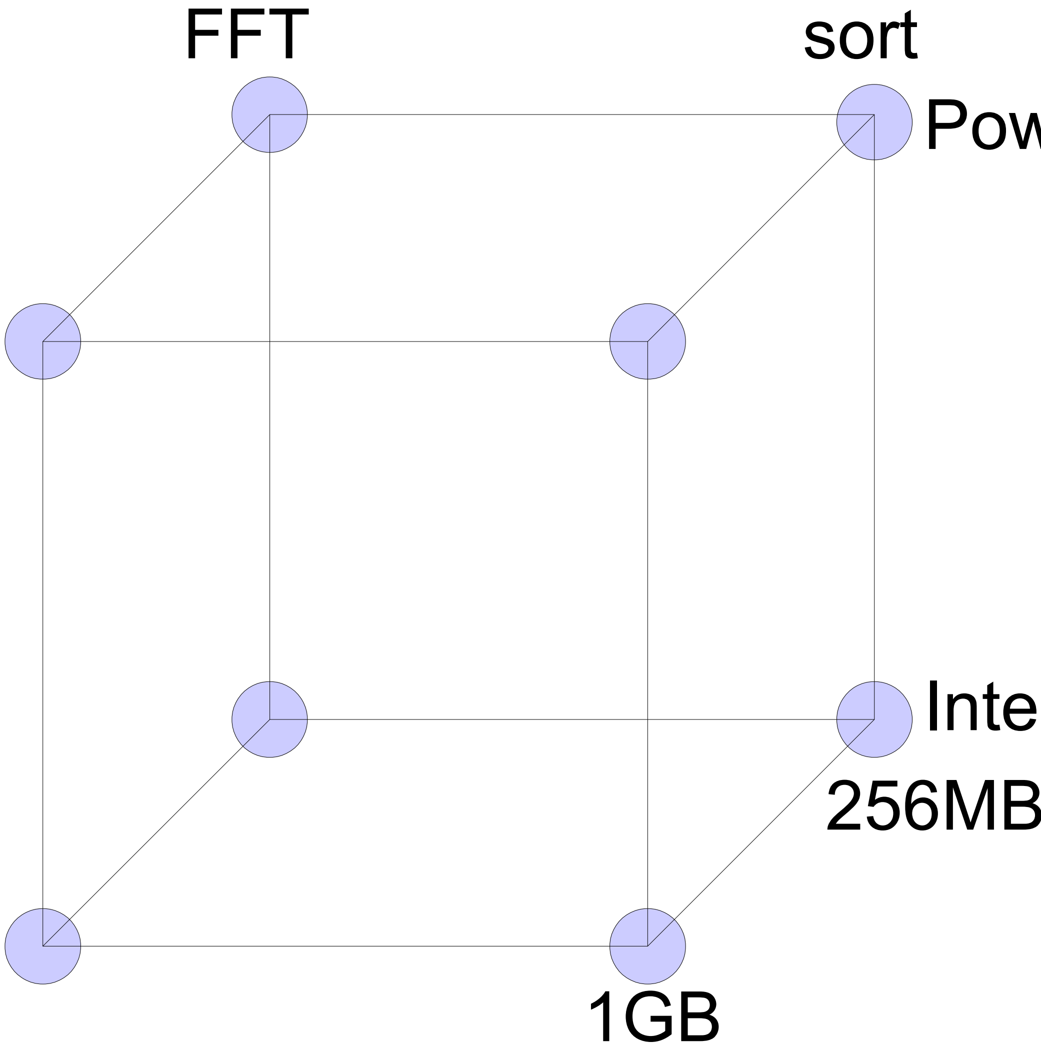
Pentium + 1GB + sort

Full factorial design

- Measure all possible combinations of levels of the different factors
- With k factors and n_i levels, the number of experiments is

$$\prod_i^k n_i$$

- Provides full information about all interactions at the price of more work

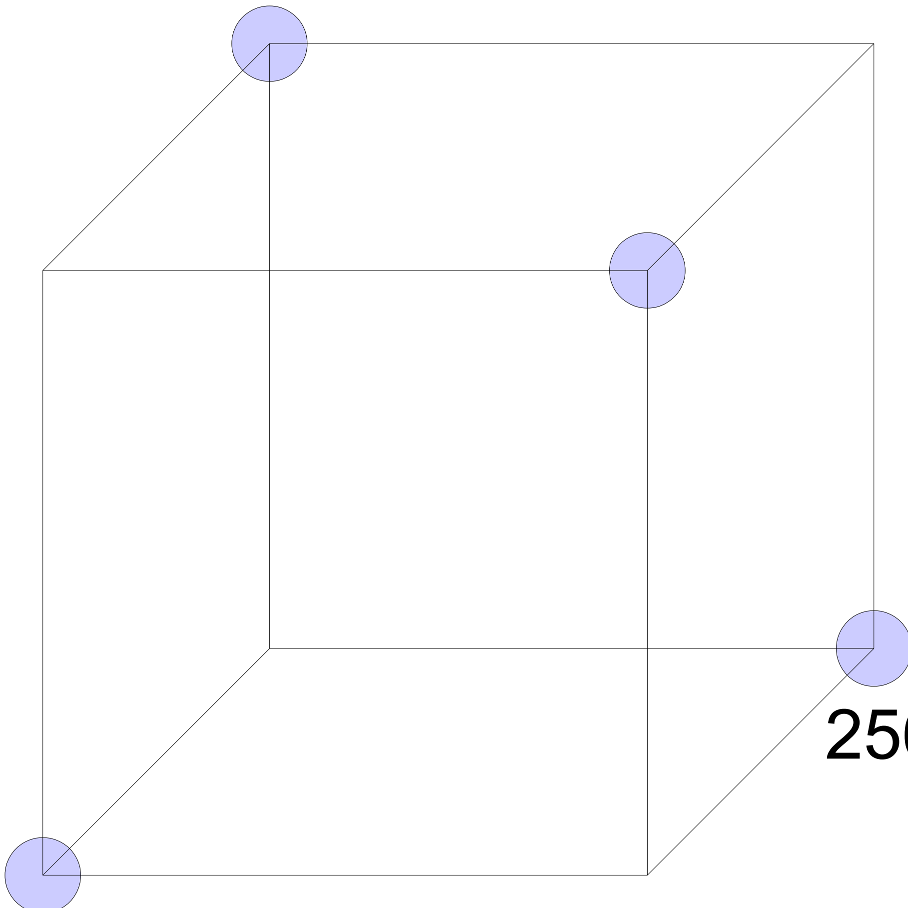


PowerPC + 256MB + sort
PowerPC + 256MB + FFT
PowerPC + 1 GB + sort
PowerPC + 1 GB + FFT

Pentium + 256MB + sort
Pentium + 256MB + FFT
Pentium + 1 GB + sort
Pentium + 1 GB + FFT

Fractional factorial design

- Measure a subset of the possible combinations
- Attempt to obtain the most information for the minimal work
- Will be able to identify some interactions
- But cannot distinguish sets of interactions



Pentium + 256MB + sort

Pentium + 1 GB + FFT

PowerPC + 256MB + FFT

PowerPC + 1 GB + sort

- Common designs
- 2^k design: a full design of *k* factors with 2 levels each
- $2^k r$ design: same as 2^k, but each experiment is repeated *r* times
- 2^{k-p} design: a partial design with *k* factors but performing less experiments

Analyzing a 2^2 Design

2^2: 2 factors, each has 2 levels

- For example, the first factor can be memory
  - Call this factor X_A
  - Its levels are 256MB and 1GB
- Let the second factor be cache size
  - Call this factor X_B
  - Let its levels be 16KB and 32KB

The levels need to span the relevant range (this may change with time / technology)

- Perform a full factorial design, that is measure all 4 combinations
- Results can be shown in a table:

		memory	
		256MB	1GB
Cache	16KB	15	45
	32KB	25	75

- Assume a model with 4 unknowns:

$$y = q_0 + q_A X_A + q_B X_B + q_{AB} X_A X_B$$

Abstracting the results

(with levels ±1)

		X_A	
		-1	1
X_B	-1	y_1	y_2
	1	y_3	y_4

And model

$$y = q_0 + q_A X_A + q_B X_B + q_{AB} X_A X_B$$

Leads to 4 equations with 4 unknowns

$$y_1 = q_0 - q_A - q_B + q_{AB}$$
$$y_2 = q_0 + q_A - q_B - q_{AB}$$
$$y_3 = q_0 - q_A + q_B - q_{AB}$$
$$y_4 = q_0 + q_A + q_B + q_{AB}$$

$$y_1 = q_0 - q_A - q_B + q_{AB}$$
$$y_2 = q_0 + q_A - q_B - q_{AB}$$
$$y_3 = q_0 - q_A + q_B - q_{AB}$$
$$y_4 = q_0 + q_A + q_B + q_{AB}$$

Summing them up leads to

$$y_1 + y_2 + y_3 + y_4 = 4\,q_0 + 0\,q_A + 0\,q_B + 0\,q_{AB}$$

that is, $q_0 = \frac{1}{4}\left(y_1 + y_2 + y_3 + y_4\right)$

Similar algebraic manipulations lead to the solutions

$$q_0 = \frac{1}{4}\left(y_1 + y_2 + y_3 + y_4 \right)$$

$$q_A = \frac{1}{4}\left(-y_1 + y_2 - y_3 + y_4 \right)$$

$$q_B = \frac{1}{4}\left(-y_1 - y_2 + y_3 + y_4 \right)$$

$$q_{AB} = \frac{1}{4}\left(y_1 - y_2 - y_3 + y_4 \right)$$

For the example

		memory	
		256MB	1GB
Cache	16KB	15	45
	32KB	25	75

This procedure leads to the model

$$y = 40+20X_A+10X_B+5X_AX_B$$

(that is $q_0=40, q_A=20, q_B=10, q_{AB}=5$)

A sign table can be used for the computation:

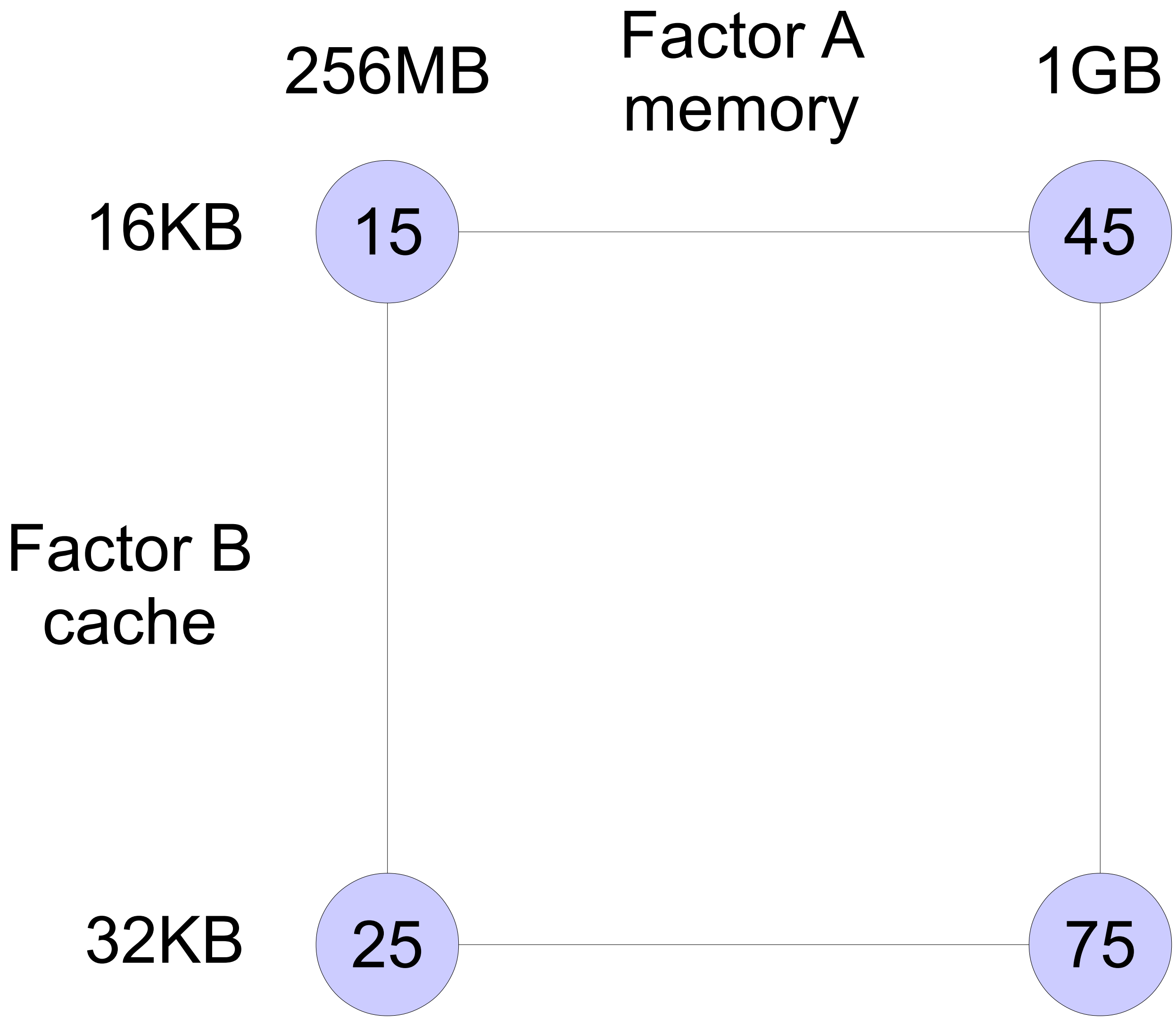
	I	A	B	AB	Y
	1	-1	-1	1	15
	1	1	-1	-1	25
	1	-1	1	-1	45
	1	1	1	1	75
Sum	160	80	40	20	
Sum/4	40	20	10	5	
	q_0	q_A	q_B	q_{AB}	

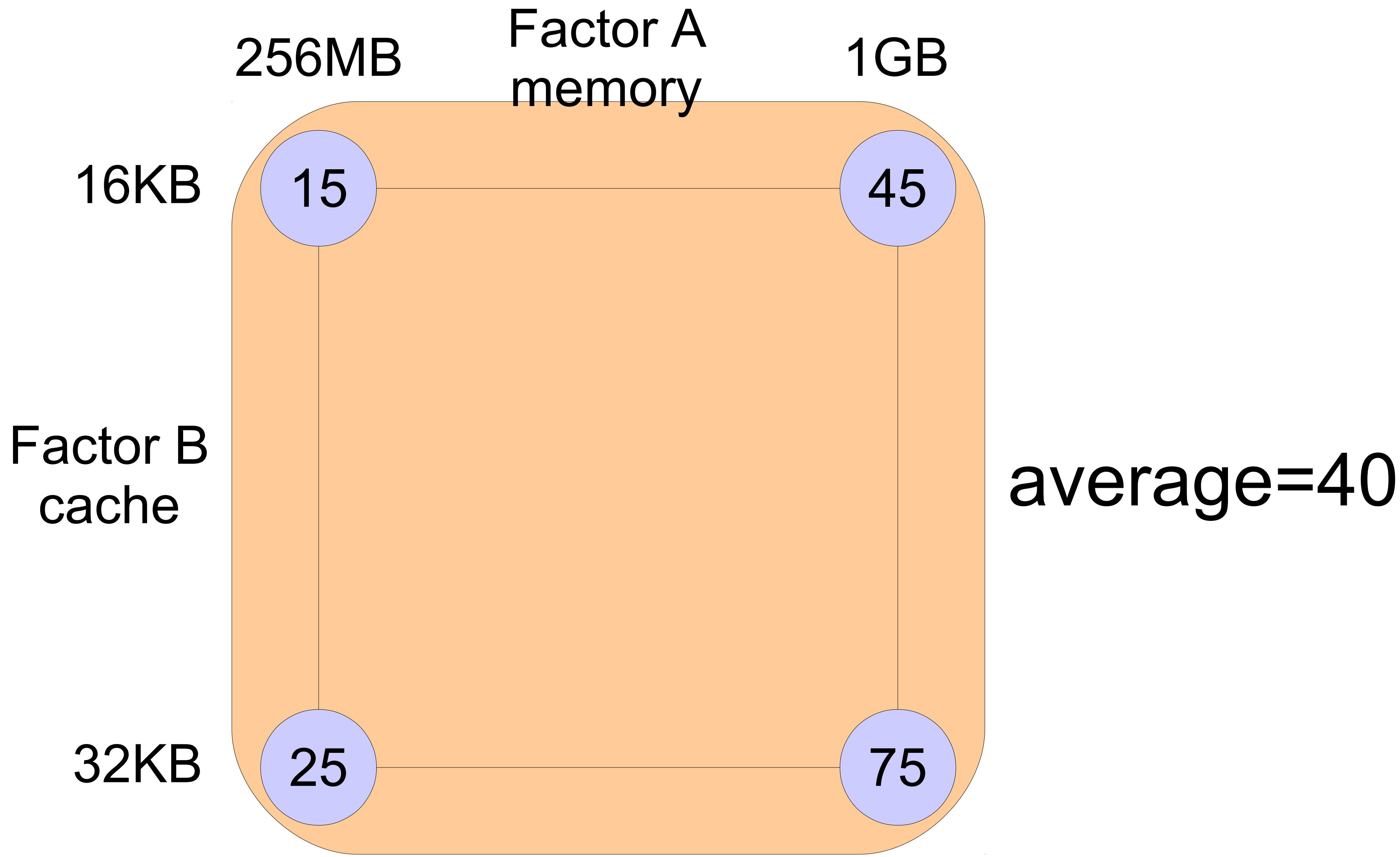
For larger designs, find the appropriate table, plug in the results, and get the answer

Interpretation of these results:

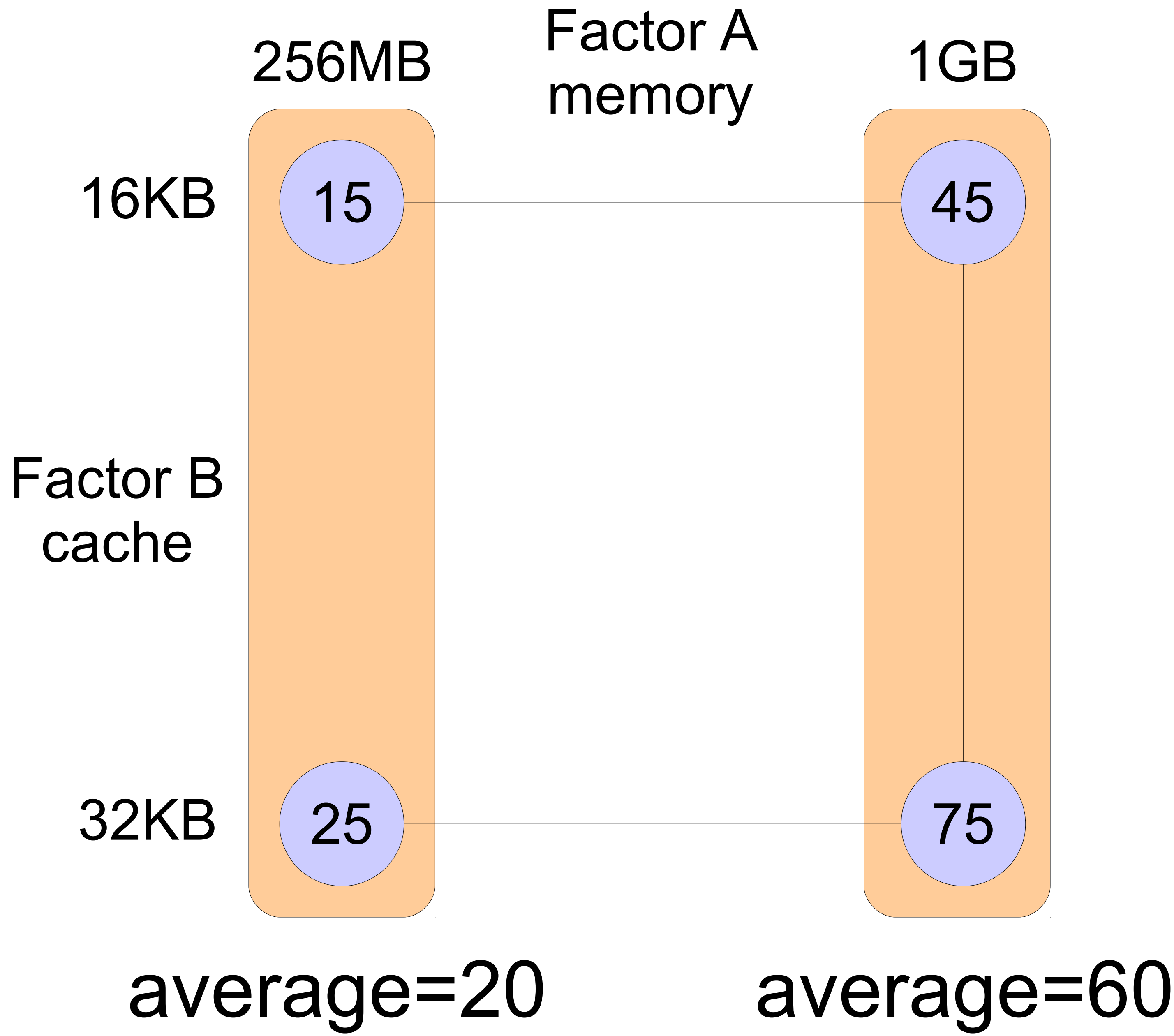
- q_0=40 : the average of all 4 measurements is 40
- q_A=20 : the memory factor has an effect of ±20
- q_B=10 : the cache factor has an effect of ±10
- q_{AB}=5 : the interaction has an effect of ±5

Each of these effects is an average over all the levels of the other factors

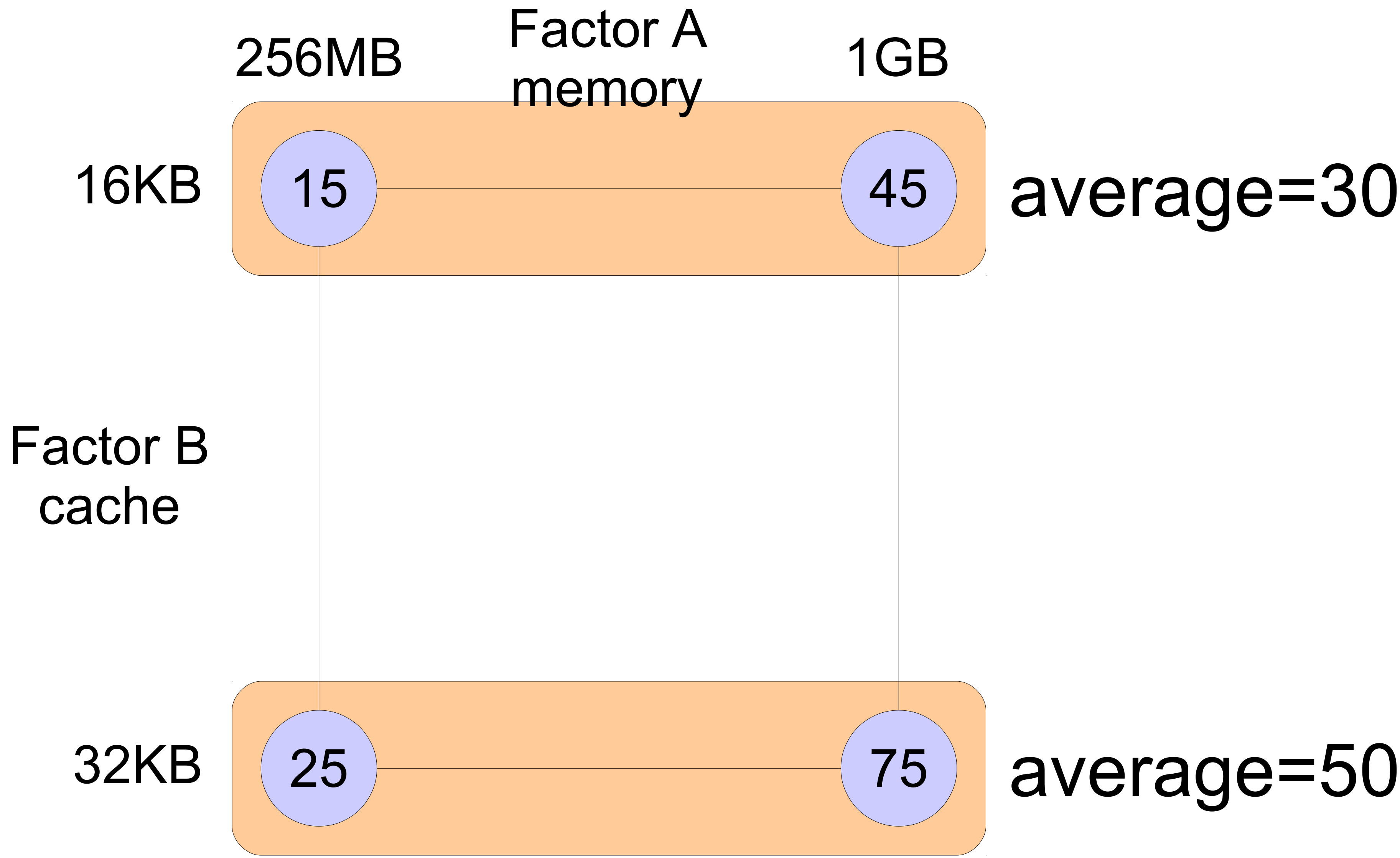

Factor A
memory
256MB
1GB
Factor B
cache
16KB
32KB
15
45
25
75



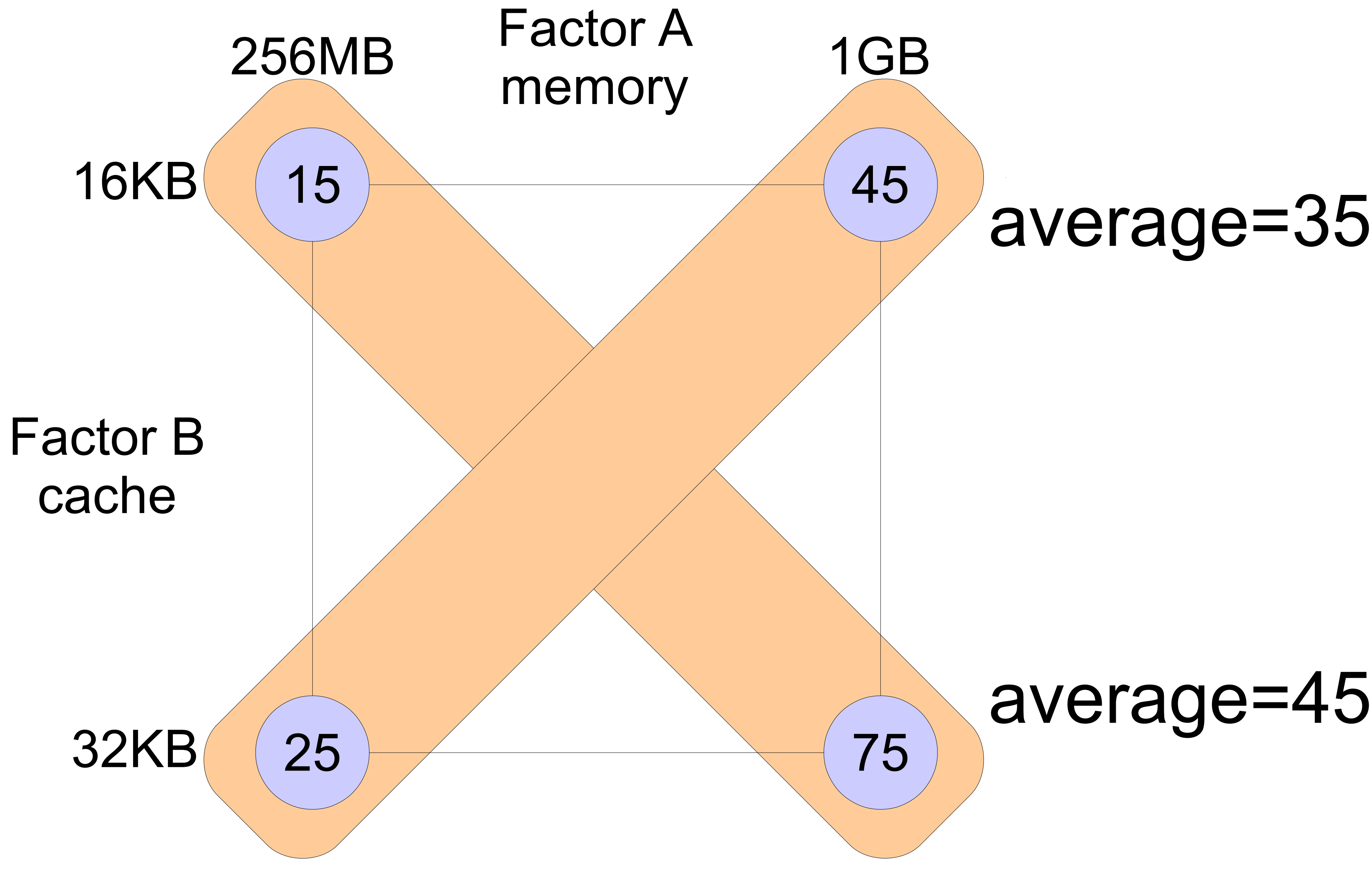
q_0=40 : the average of all 4 measurements is 40



q_A=20 : the memory factor has an effect of ±20



q_B=10 : the cache factor has an effect of ±10



q_{AB}=5 : the interaction has an effect of ±5

What do interactions mean?

- Consider two balanced systems
  - The CPU and I/O subsystem are both adequate
- Or alternatively, two unbalanced systems
  - Fast CPU and slow I/O
  - Slow CPU and fast I/O
- Evaluate them using two programs
  - A compute-intensive application
  - An I/O-intensive application

Balanced Systems	application CPU	I/O
CPU A I/O A	37	37
CPU B I/O B	35	43

Unbalanced Systems	application CPU	I/O
Fast CPU Slow I/O	10	68
Slow CPU Fast I/O	62	12

- Same q_0 (overall average)
- Same q_A (difference between left and right)
- Same q_B (difference between top and bottom)
- But very different q_{AB} (diagonals): with unbalanced systems, matching the benchmark to the system is meaningful

Allocation of variation

- SST = sum squares total = $\sum\left(y_i - \overline{y}\right)^2$
- SSA = 4 q_A^2
- SSB = 4 q_B^2
- SSAB = 4 q_{AB}^2
- Surprise: SST = SSA + SSB + SSAB

Explanation:

$$\begin{aligned}\sum\left(y_i-\bar{y}\right)^2 &= \sum\left(q_A x_A+q_B x_B+q_{AB} x_A x_B\right)^2\\ &= \sum\left(q_A x_A\right)^2+\sum\left(q_B x_B\right)^2\\ &\quad +\sum\left(q_{AB} x_A x_B\right)^2\\ &\quad +cross\ terms\end{aligned}$$

the cross terms cancel out because the x's are ±1 in all possible combinations

$$\sum\left(q_A x_A\right)^2=q_A^2\sum x_A^2=4\,q_A^2$$

Allocation of variation

- SST = sum squares total = $\sum \left(y_i - \bar{y} \right)^2$
- SSA = 4 q_A^2
- SSB = 4 q_B^2
- SSAB = 4 q_{AB}^2
- Surprise: SST = SSA + SSB + SSAB
- So we can allocate the part of the variation due to each factor and to the interaction:

$$\frac{SSA}{SST} \qquad \frac{SSB}{SST} \qquad \frac{SSAB}{SST}$$

Example

$$\bar{y} = q_0 = 40$$

		memory	
		256MB	1GB
Cache	16KB	15	45
	32KB	25	75

$$\begin{aligned} SST &= (15-40)^2 + (45-40)^2 + (25-40)^2 + (75-40)^2 \\ &= (-25)^2 + 5^2 + (-15)^2 + 35^2 \\ &= 2100 \end{aligned}$$

$$\frac{SSA}{SST} = \frac{4 \cdot 20^2}{2100} = \frac{1600}{2100} = 76\%$$

$$\frac{SSB}{SST} = \frac{4 \cdot 10^2}{2100} = \frac{400}{2100} = 19\%$$

$$\frac{SSAB}{SST} = \frac{4 \cdot 5^2}{2100} = \frac{100}{2100} = 5\%$$

Reservations

- The relative importance of the different factors is exaggerated due to squaring
- The values depend on the specific measurements, which depend on the specific levels used
- Also depends on the model

An alternative: a multiplicative model

- Take the log of the results before analyzing

$$\ln(y_i) = q_0 + q_A X_A + q_B X_B + q_{AB} X_A X_B$$

- The model then becomes

$$y_i = e^{q_0} \cdot e^{q_A X_A} \cdot e^{q_B X_B} \cdot e^{q_{AB} X_A X_B}$$

- The choice of model should depend on an understanding of the domain
- In particular, a multiplicative model is appropriate if the combined effect of the factors is expected to be multiplicative
- Example:
  - Factor A is the CPU speed (or slowness) in cycles-per-instruction
  - Factor B is the program length
  - Execution time is their product
- A high interaction (q_{AB}) may indicate that a multiplicative model should be checked

Fractional Design

- A full factorial design with 7 factors and 2 levels requires 2^7=128 experiments
- A fractional design like 2^{7-4} can reduce this to a much lower number: 2^3=8
- The question is how to select the combinations of levels to use
- The answer: try to reduce “confounding"

General procedure for 2^{k-p} fractional design:

- Create a sign table for a 2^d full design, where $d=k-p$
  - This has one column of all 1s
  - d columns for the d factors
  - $2^d - d - 1$ columns of interactions
- Use the d factor columns for the first d factors
- Use k-d of the interaction columns for the remaining factors
- Set the factor levels in each experiment (line) according to the signs of the different factor columns

Example: a 2^{7-4} fractional design

The sign table for a 2^3 full design is

I	A	B	C	AB	AC	BC	ABC
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

Replace 4 interactions with the missing factors

ones

factors

interactions

Example: a 2^{7-4} fractional design

The new sign table is

I	A	B	C	D	E	F	G
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

ones

factors

Here we use all the interaction columns

- The problem: confounding
- Each column no longer represents a single factor or interaction
  - Example: the last column was ABC, and now it has the added role of G, and a few others
- With 7 factors, there are 128 q's representing factors and interactions
- But we only make 8 measurements
- So each one represents the combined effect of 16 factors and interactions!

Another example: a 2^{4-1} fractional design

The sign table for a 2^3 full design is

I	A	B	C	AB	AC	BC	ABC
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

ones

factors

interactions

Need to replace only 1 interaction

Another example: a 2^{4-1} fractional design

Let's select the ABC column

I	A	B	C	AB	AC	BC	D
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

But each column actually represents two effects!

The confounding in this example is

I	A	B	C	AB	AC	BC	D
ABCD	BCD	ACD	ABD	CD	BD	AD	ABC
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

Average confounded with 4th order interaction

2nd order interactions confounded with each other

Main effects confounded with 3rd order interactions

Assuming that higher order interactions are weaker, this is good

But if we select the AB column

I	A	B	C	D	AC	BC	ABC
ABD	BD	AD	ABCD	AB	BCD	ACD	CD
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

Average confounded with 3rd order interaction

Some main effects confounded with 2nd order interactions

Assuming that lower order interactions are stronger, this is worse

But how do we find the confoundings?

- Columns of interactions are derived by point multiplication of the columns of the effects
- So need to find all the different combinations that give the same result

I	A	B	C	AB	AC	BC	D
ABCD	BCD	ACD	ABD	CD	BD	AD	ABC
1	-1	-1	-1	1	1	1	-1
1	1	-1	-1	-1	-1	1	1
1	-1	1	-1	-1	1	-1	1
1	1	1	-1	1	-1	-1	-1
1	-1	-1	1	1	-1	-1	1
1	1	-1	1	-1	1	-1	-1
1	-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1	1

Repeated measurements

- A $2^k r$ design implies *r* repetitions of each experiment
- This enables an assessment of the experimental error
- And calculation of confidence intervals for the q's

The model with an error is

$$y = q_0 + q_A X_A + q_B X_B + q_{AB} X_A X_B + e$$

- The average result of each experiment is

$$\bar{y}_i = \frac{1}{r}\sum_j y_{ij}$$

And this is used to calculate the effects

- In addition, we have the errors

$$e_{ij} = y_{ij} - \bar{y}_i$$

- By definition, the sum of errors in each experiment is 0
- But the sum of the squares of the errors is not

$$SSE = \sum_i \sum_j e_{ij}^2$$

- The fraction of the variation due to errors is then $\dfrac{SSE}{SST}$

- To calculate confidence intervals, we need a model of the variance of each effect
- Assume that the y_{ij}s are normally distributed with variance σ^2
- q_0 is the sum of many such random variables

$$q_0 = \frac{1}{4r} \sum_{ij} y_{ij}$$

- So it is also normally distributed, with variance

$$\frac{\sigma^2}{4r}$$

- Empirically, this variance is related to the variation allocated to the error

$$s_e^2 = \frac{SSE}{4(r-1)}$$

- Therefore the estimate for the variance of q_0 is

$$s_{q_0}^2 = \frac{s_e^2}{4\mathrm{r}}$$

- And the confidence interval is

$$q_0 \pm t_{1-\frac{\alpha}{2},\,4(r-1)} \cdot \frac{s_e}{\sqrt{4\mathrm{r}}}$$