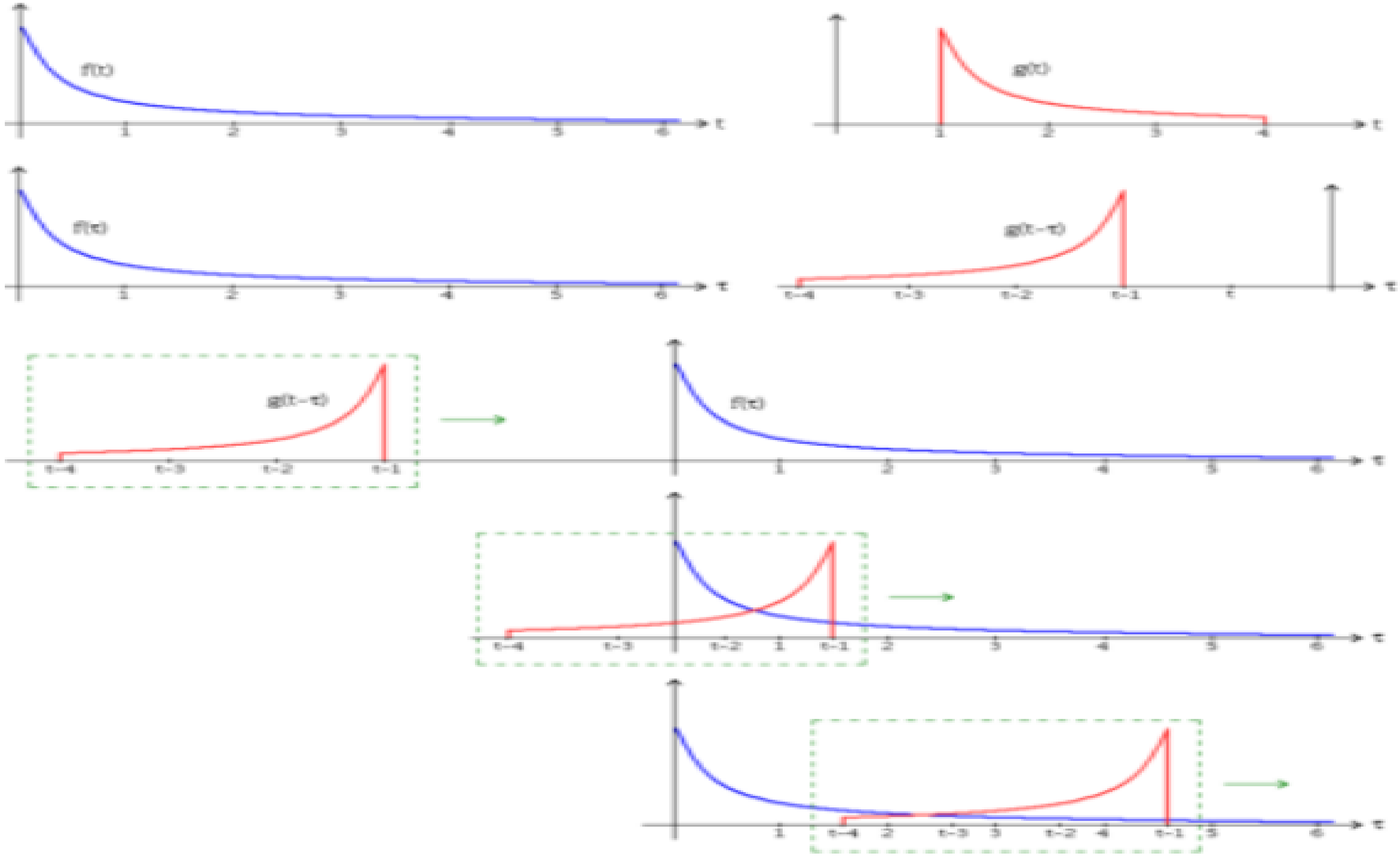
PARALLELIZED CONVOLUTION

Convolution

- Convolution is a mathematical operation on two functions
- A function derived from two given functions by integration that expresses how the shape of one is modified by the other.
- The Mathematical expression for basic two dimensional convolution is

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

Convolution



Applications

- Image Processing
- Electrical Engineering (Communication signal processing)
- Statistics
- Differential equations

Matrix Convolution

I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	I_{16}	I_{17}	I_{18}	I_{19}
I_{21}	I_{22}	I_{23}	I_{24}	I_{25}	I_{26}	I_{27}	I_{28}	I_{29}
I_{31}	I_{32}	I_{33}	I_{34}	I_{35}	I_{36}	I_{37}	I_{38}	I_{39}
I_{41}	I_{42}	I_{43}	I_{44}	I_{45}	I_{46}	I_{47}	I_{48}	I_{49}
I_{51}	I_{52}	I_{53}	I_{54}	I_{55}	I_{56}	I_{57}	I_{58}	I_{59}
I_{61}	I_{62}	I_{63}	I_{64}	I_{65}	I_{66}	I_{67}	I_{68}	I_{69}

K_{11}	K_{12}	K_{13}
K_{21}	K_{22}	K_{23}

$$O_{57} = I_{57}K_{11} + I_{58}K_{12} + I_{59}K_{13} + I_{67}K_{21} + I_{68}K_{22} + I_{69}K_{23}$$

$$O(i,j) = \sum_{k=1}^{m}\sum_{l=1}^{n} I(i+k-1, j+l-1)K(k,l)$$

Sequential solution

- The kernel matrix is padded over the input matrix and the overlapping pixels are computed and this operation is continued for all the pixels of input matrix.
- The complexity is similar to matrix multiplication where the computation involves huge no of multiplications and additions.

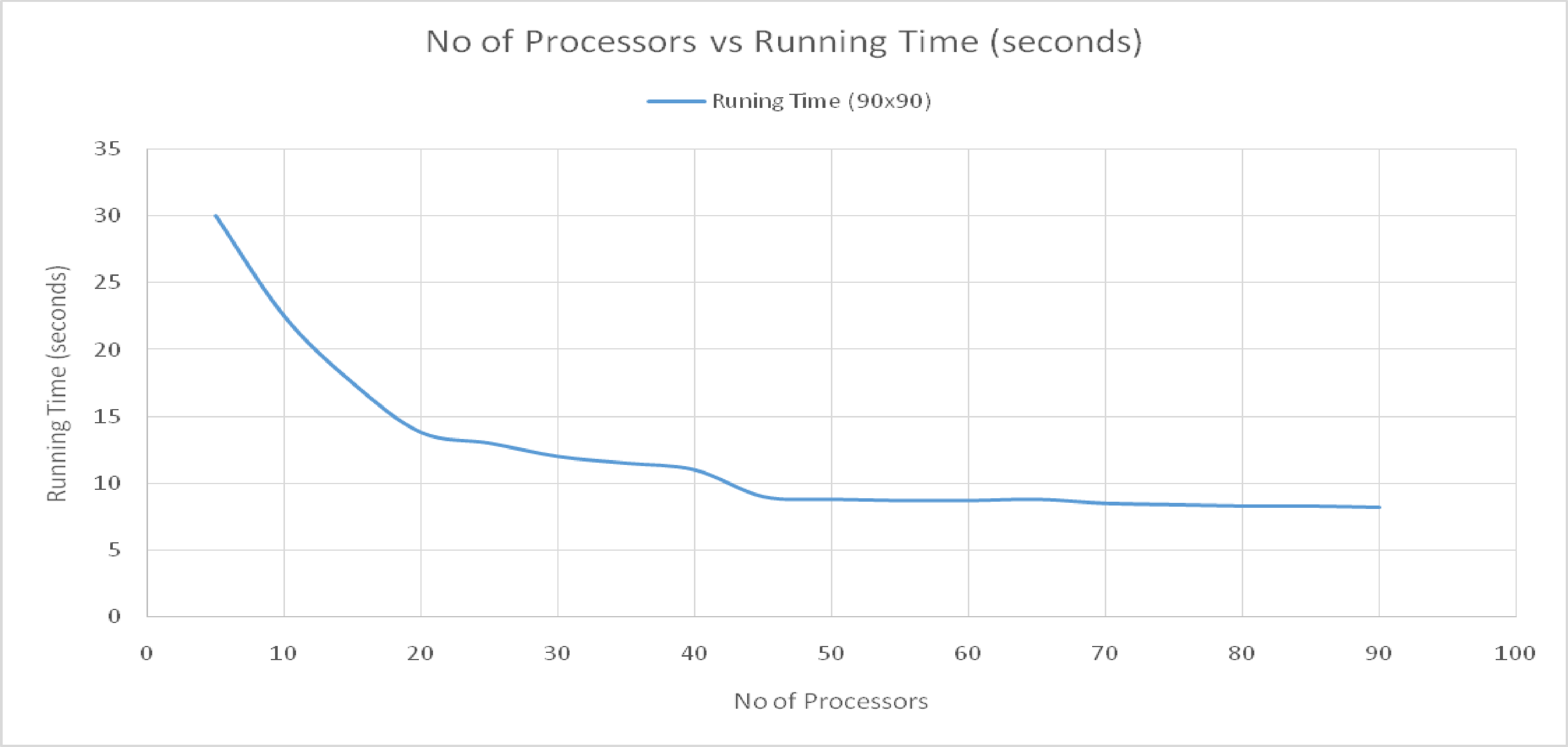
Parallel solution

- Input Matrix is stored in a single node
- Kernel matrix is broadcasted to all the other processors
- Input Matrix node distributes chunks of input matrix to all the other processors
- All the processors sends the partially computed result to a single final node
- Because of independent convolutions, distributed parallelism can be implemented

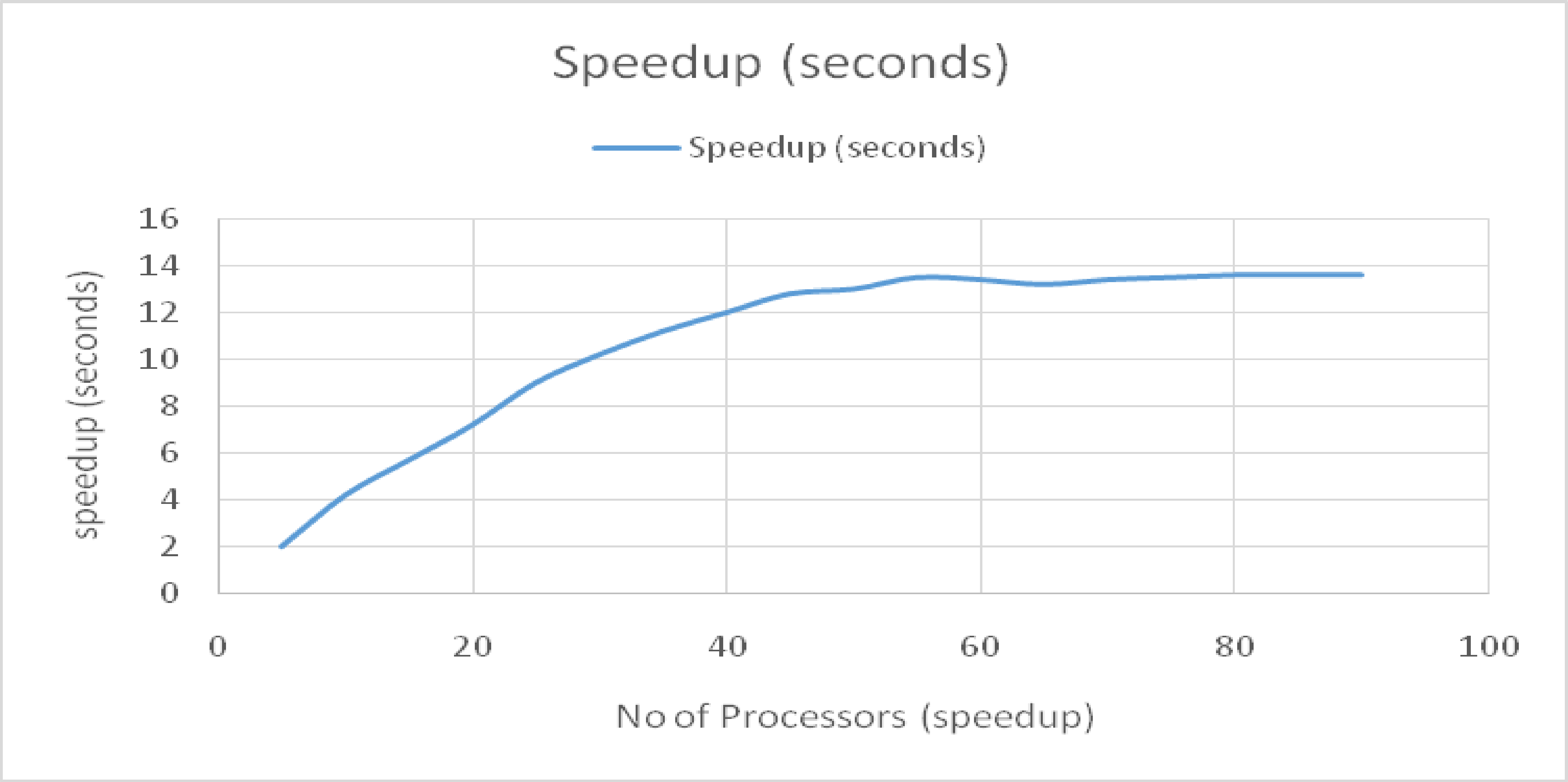
The Input size

- 1024 x 1024 image – input
- 90 x 90 kernel

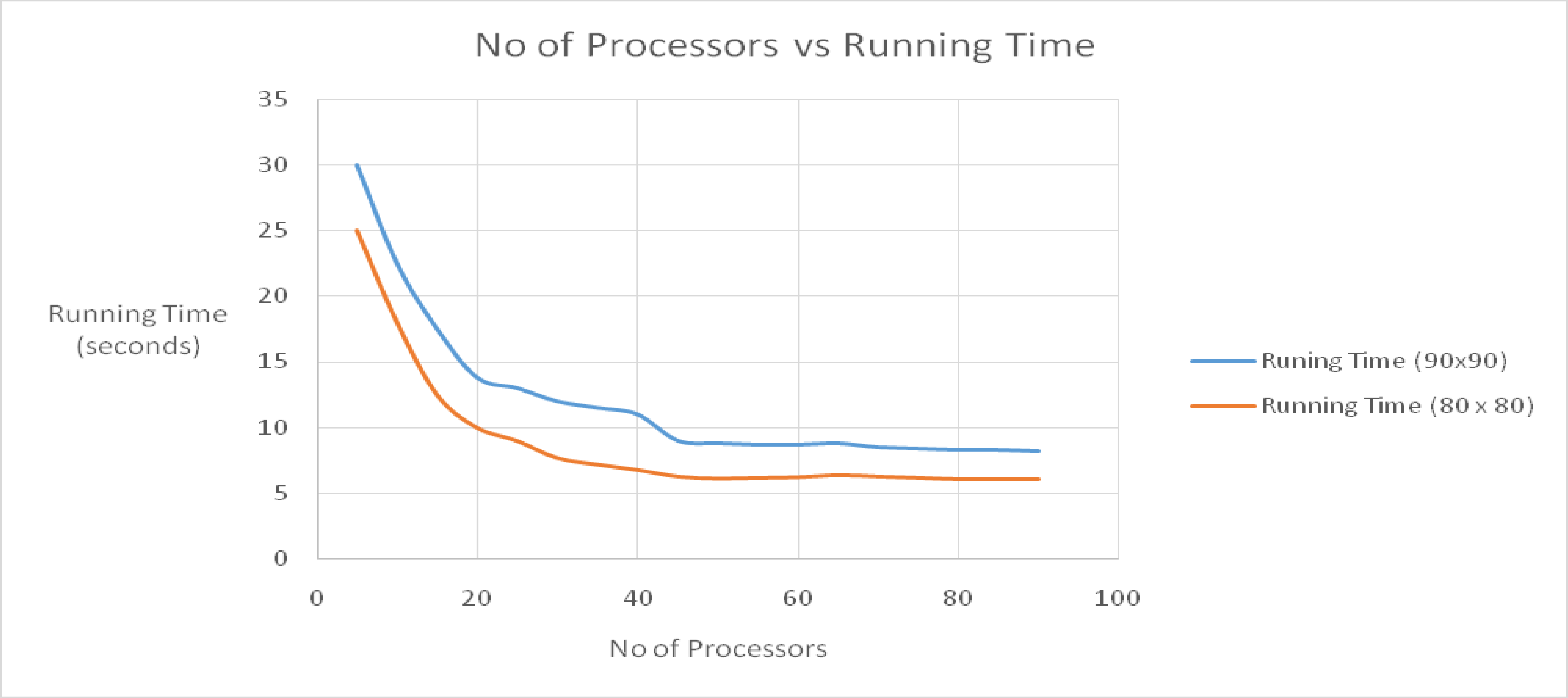
Running Time



No of Processors vs Speedup



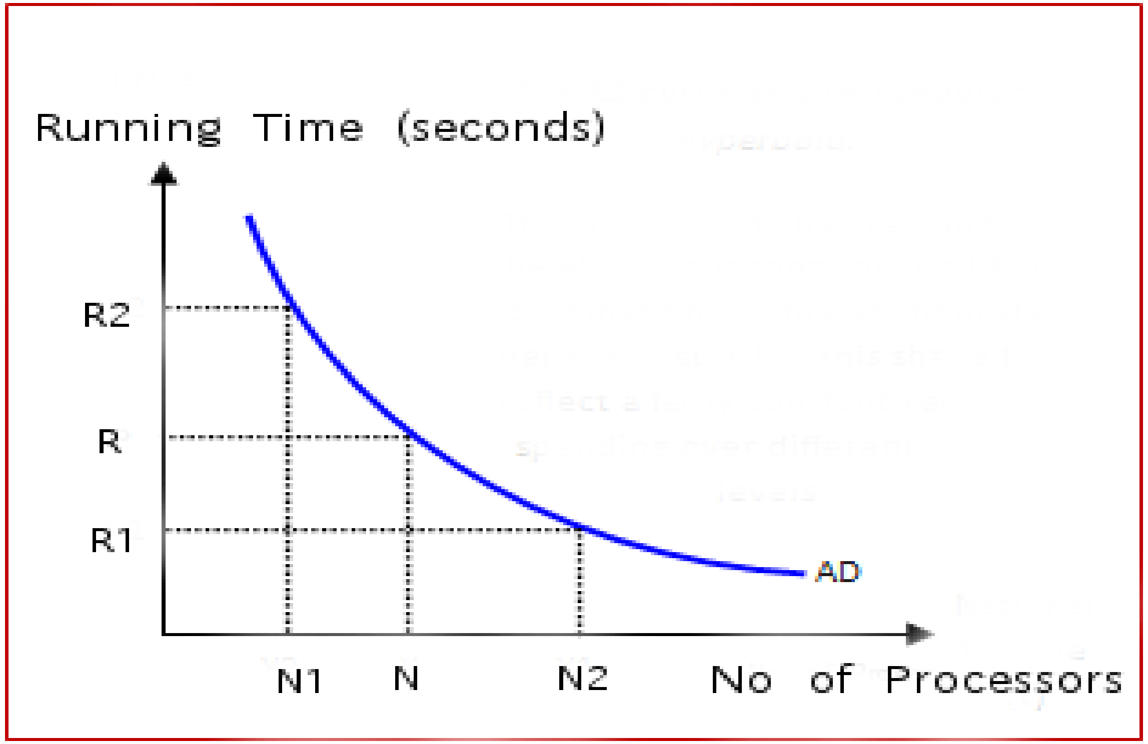
Running Time



Running Time

No of Processors allotted	Processors Requested	Running Time (seconds)	Analysis
80	160	92.12	
80	135	87.34	
80	115	83.98	
80	90	79.63	
80	80	76.38	Best Running Time
52	96	90.54	
52	84	87.91	
52	75	82.25	
52	63	76.54	
52	52	73.68	Best Running Time

Running Time



Choosing optimum N from the graph?

xy=constant

x+y=minimum

x	y	cost
1	64	64
2	32	64
4	16	64
8	8	64
16	4	64
32	2	64
64	1	64

8+8=16 - which is the minimum among all x+y combinations. so choosing N (no of processors) at this point will give fairly best cost for a given input

Optimum N?

- The optimum no of nodes for the approximate no of multiplications can be calculated.
- example : 1024 x 1024 image – input

  90 x 90 kernel

  - 1024 x 1024 x 90 x 90 = ~ 8 billion operations

Difficulties

- Communication overhead
- Large no of multiplications
  - (~8 billion Multiplications and additions) for 90 x 90 kernel
  - (~6.5 billion Multiplications and additions) for 80 x 80 kernel
- Filtering the input matrix

References

- www.scribd.com/doc/58013724/10-MPI-programmes
- http://heather.cs.ucdavis.edu/~matloff/mpi.html
- Miller, Russ, and Laurence Boxer. Algorithms, sequential & parallel: A unified approach.

Questions ?