1. Theoretical Analysis

The cost for Strassen's algorithm and the standard matrix multiplication are, respectively, $T_s(n) = 7T_s(n/2) + 18(n/2)^2$ and $T_c(n) = n^2(2n-1)$. To theoretically arrive at an optimal crossover-point, I substituted the cost of the standard matrix multiplication, T_c, in the recurrence for Strassen's and set $T_s(n) = T_c(n)$:

$$T_c(n) = 7T_c(n/2) + 18(n/2)^2$$
$$n^2(2n-1) = 7(n/2)^2(2(n/2)-1) + 18(n/2)^2$$

With this, for even n I solved the equation:

$$n^2(2n-1) = 7(n/2)^2(n^2-1) + 18(n/2)^2$$

$n = 15$ is a solution, in which case our crossover point would be 14 because n is even. . For odd n, we must have n=2k-1, so we solve the equation:

7(k+1)^2(2k+1)+18(k+1)^2 =(2k+1)^2(2(2k+1)−1)

Here, k is about equal to eighteen so our crossover point would be 37.

2. Experiment and Data

To experimentally determine the crossover point for Strassen's algorithm, I chose different crossover points based on the (n/2) division process within the algorithm and tested the running time for them on different sized matrices. For even sized matrices, I found that it appeared a crossover point of 64 was optimal, as shown by the running time getting larger and then smaller again. My final estimation for a good cutoff point was 100 based on this, accounting for the fact that it will need to be higher for odd matrices.

	64	128	256	512
n=512				0.676
n=256			0.085	0.589
n=128		0.0104	0.068	0.481
n=64	0.001	0.0120	0.101	0.540
n=32	0.002	0.0148	0.100	0.739

Using this crossover point, Strassen’s algorithm was always faster than the standard matrix multiplication method:

	standard	strassen
512	0.772	0.339
256	0.086	0.077
128	0.012	0.009
64	0.001	0.001
32	0.000	0.000