TASK #1:

Question #1:

M = Guests
N = Tables
F = Number of pairs of friends
E = Number of pairs of enemies

Atleast One Table Constraint:
Number of Clauses: **M**

Each guest i should be seated in atleast one table (for each i that $1 \le i \le M$)

FOL: $\neg X_{i1} \land \neg X_{i2} \ldots \land \neg X_{i(n-1)} \land \neg X_{i(n+1)} \ldots \land \neg X_{iN} \Rightarrow X_{in}$ (for each n that $1 \le n \le N$)
CNF: $\neg [\neg X_{i1} \land \neg X_{i2} \ldots \land \neg X_{i(n-1)} \land \neg X_{i(n+1)} \ldots \land \neg X_{iN}] \lor X_{in}$ (for each n that $1 \le n \le N$)
which is equivalent to $X_{i1} \lor X_{i2} \ldots \lor X_{iN}$ (for each n that $1 \le n \le N$)

Atmost One Table Constraint:
Number of Clauses: **M * (N * (N-1))/2**

Each guest i should be seated in atmost one table. i.e If the guest is seated in table k. He should not be seated in other tables. (for each i that $1 \le i \le M$)

FOL: $X_{ik} \Rightarrow \neg X_{in}$ (for each k and n that $1 \le k \ne n \le N$)
CNF: $\neg X_{ik} \lor \neg X_{in}$

Friends Constraint:
Number of Clauses: **2 * F * N**

Guests i and j who are friends should be seated in same table.

FOL: $X_{in} \Leftrightarrow X_{jn}$ (for each n that $1 \le n \le N$)
CNF: $[X_{in} \Rightarrow X_{jn}] \land [X_{jn} \Rightarrow X_{in}]$
$[\neg X_{in} \lor X_{jn}] \land [\neg X_{jn} \lor X_{in}]$

In KB, The above CNF will be added as two clauses.

Enemies Constraint:
Number of Clauses: **E * N**

Guests i and j who are enemies should be seated in different table.

FOL: $[X_{in} \Rightarrow \neg X_{jn}] \land [X_{jn} \Rightarrow \neg X_{in}]$ (for each n that $1 \le n \le N$)
CNF: $[\neg X_{in} \lor \neg X_{jn}] \land [\neg X_{jn} \lor \neg X_{in}]$ which is equivalent to $[\neg X_{in} \lor \neg X_{jn}]$

Total number of clauses: M + (M * (N * (N-1))/2) + (2 * F * N) + (E * N)

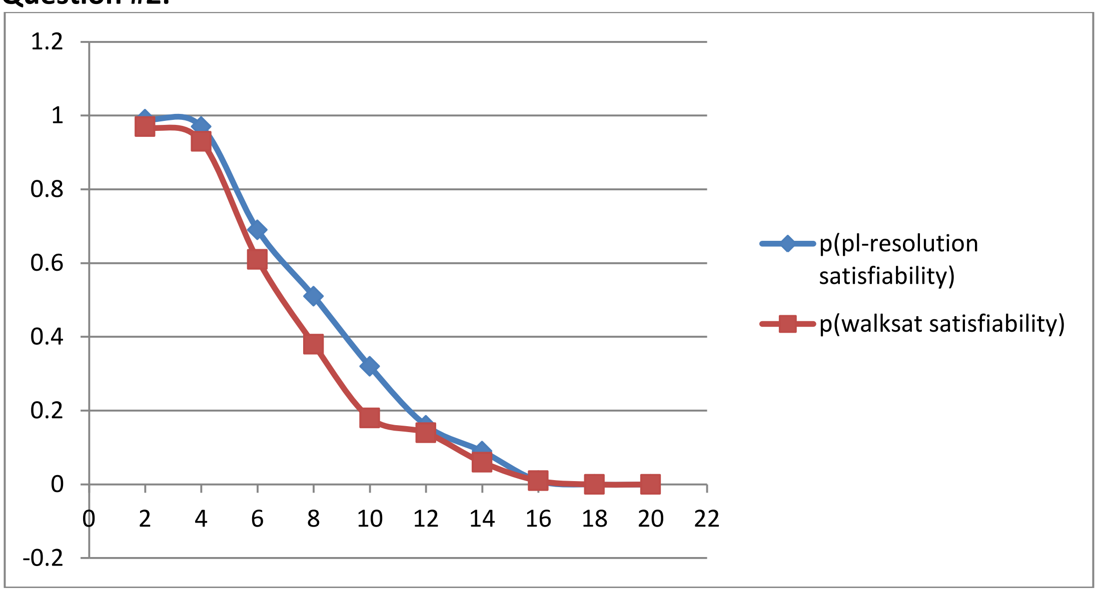
TASK #4:

Experiment #1:

probability: 50%

Guests	Tables	f-percent	e-percent	p(pl-resolution satisfiability)	p(walksat satisfiability)
16	2	0	2	0.99	0.97
16	2	0	4	0.97	0.93
16	2	0	6	0.69	0.61
16	2	0	8	0.51	0.38
16	2	0	10	0.32	0.18
16	2	0	12	0.16	0.14
16	2	0	14	0.09	0.06
16	2	0	16	0.01	0.01
16	2	0	18	0	0
16	2	0	20	0	0

Question #2:



X – Axis: e-percent
Y - Axis: P(Satisfiability) Scale

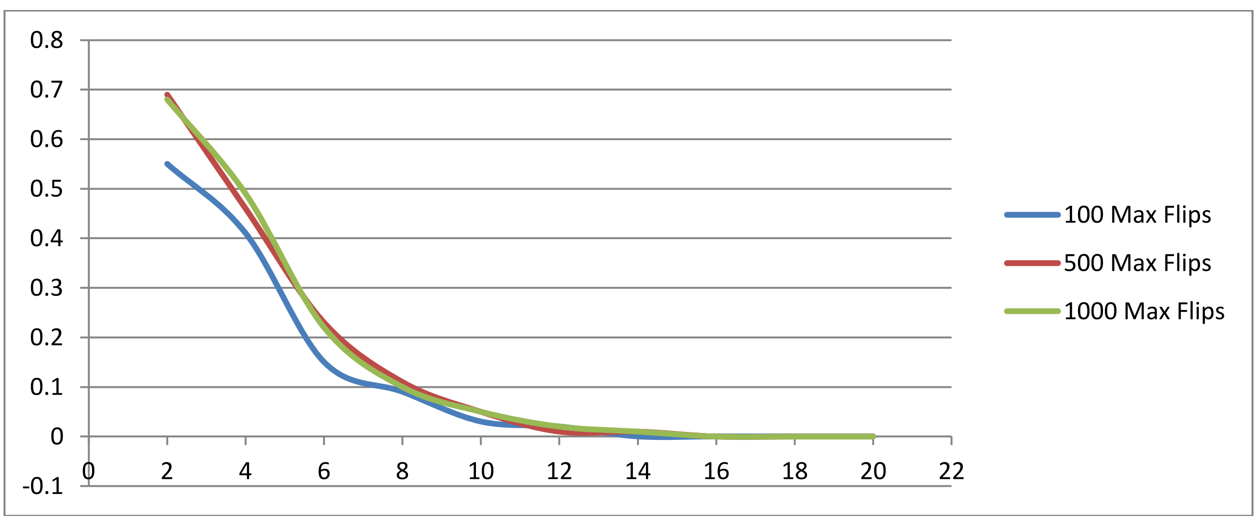
They are not same. P(PL-Resolution Satisfiability) is greater than P(Walk SAT Satisfiability) because the walk sat algorithm will not be able to identify the satisfiability with max flips set at 100. (i.e PL-Resolution is a sound and complete algorithm that will determine the satisfiability with certainity. But, Walk SAT algorithm is bound by Max Flip value to determine the satisfiability. It cannot absolutely determine unsatisfiability). If we increase the max flips count, then the satisfiability count of Walk SAT may increase. For greater than 14%, both the algorithms do not find satisfiability. So, P(pl-resolution) and P(walk sat) is close to zero.

Experiment #2:

Walk SAT probability : 50%

Guests	Tables	e-percent	f-percent	p(100 Max Flips Satisfiability)	p(500 Max Flips Satisfiability)	p(1000 Max Flips Satisfiability)
16	2	5	2	0.55	0.69	0.68
16	2	5	4	0.41	0.46	0.49
16	2	5	6	0.15	0.23	0.22
16	2	5	8	0.09	0.11	0.1
16	2	5	10	0.03	0.05	0.05
16	2	5	12	0.02	0.01	0.02
16	2	5	14	0	0.01	0.01
16	2	5	16	0	0	0
16	2	5	18	0	0	0
16	2	5	20	0	0	0

Chart Area:



X – Axis: f-percent
Y - Axis: P(Walk SAT Satisfiability)

Question #3:

When f-percent increases, P(Walk Sat Satisfiability) decreases. For f-percent greater than 14%, it becomes zero. It is because, with the increase in f-percent, will eventually increase the number of clauses in KB. So, Walk SAT Algorithm will not able to find the satisfiable model with the given max flips count which is <= 1000 for the 100 random instances. If we increase the max flip value, then satisfiability count may increase.

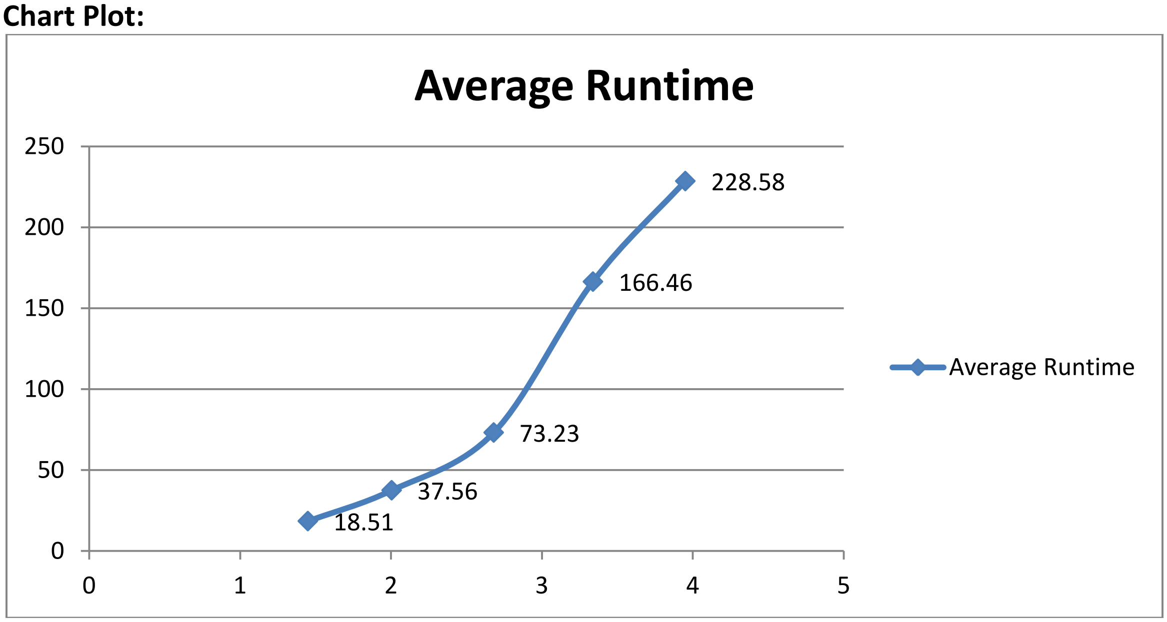
Question #4:

Based on the observation, the number of satisfiability count increases with the increase in maxflip values for 100, 500 and 1000. Based on the chart data, satisfiability value of max flip with 500 and 1000 is more than the satisfiability value of max flip with 100. Because with more max flip value, Walk SAT algorithm will need more iterations to identify the satisfiability by flipping the literals in the unsatisfied clause.

Experiment #3:

Guests	Tables	ratio(clauses/symbols)	average(runtime)
16	2	1.449375	18.51
24	3	2.004583333	37.56
32	4	2.6809375	73.23
40	5	3.33875	166.46
48	6	3.950833333	228.58

Chart Plot:



Question #5:

Yes. The ratio(Clauses/Symbol) is consistent with the equation derived from the result of Question 1. Symbols will be generated for each guest i against the tables N. Similarly, the total number of clauses is obtained by summing up all the clauses derived from Atleast One Table Constraint, Atmost One Table Constraint, Friends Constraint and Enemies Constraint.

Since the f=2% and e=2% is same for all the guests and tables arrangement. The number of clauses derived for the guests and tables in experiment should satisfy the above four constraints, which would comply with the equation from result of Question 1.

Consider the below plot table for 5 instances:

Guests	Tables	f-percent	e-percent	No of friends	No of enemies	no of iterations	no of clauses	no of symbols	ratio(clauses/symbols)
16	2	2	2	4	1	13	50	32	1.5625
16	2	2	2	0	4	8	40	32	1.25
16	2	2	2	0	2	16	36	32	1.125
16	2	2	2	2	1	14	42	32	1.3125
16	2	2	2	4	4	11	56	32	1.75

By applying the parameters in the formula of four constraints of Question 1:
Guests M = 16
Tables N = 2

<u>1st Instance:</u>
Number of pair of friends = 4
Number of pair of enemies = 1
Total Number of Clauses: 16 + 16 + 16 + 2 = 50
Ratio(Clauses/Symbols) = 50/32 = 1.5625

<u>2nd Instance:</u>
Number of pair of friends = 0
Number of pair of enemies = 4
Total Number of Clauses: 16 + 16 + 0 + 8 = 40
Ratio(Clauses/Symbols) = 40/32 = 1.25

<u>3rd Instance:</u>
Number of pair of friends = 2
Number of pair of enemies = 1
Total Number of Clauses: 16 + 16 + 8 + 2 = 42
Ratio(Clauses/Symbols) = 42/32 = 1.3125

4th Instance:
Number of pair of friends = 4
Number of pair of enemies = 4
Total Number of Clauses: 16 + 16 + 16 + 8 = 56
Ratio(Clauses/Symbols) = 56/32 = 1.75

5th Instance:
Number of pair of friends = 0
Number of pair of enemies = 2
Total Number of Clauses: 16 + 16 + 0 + 4 = 36
Ratio(Clauses/Symbols) = 36/32 = 1.125

Ratio(clauses/symbols) for the above five instances obtained by substituting the values in equation is as same as Ratio(clauses/ symbols) in the chart plot table of guests=16 and tables=2. Therefore, Ratio(clauses/symbols) obtained for all the instances will be same as ratio(clauses/symbols) by substituting the parameter values in the equation for question 1.

Question #6:

The plot with regard to Average(Runtime) and Ratio(Clause/Symbol) is consistent with the curve in book AIMA Figure 7.19(b). In the curve plot, The maximum of ratio(clause/symbol) for the size of wedding guests=48 and tables=6 is upto 3.95 which contains the average(runtime) = [Close to 230]. Therefore, with the increase in size of wedding, the number of iterations to find the 100 satisfied sentences also increases. Since the max flips is set as 1000. Higher the wedding plan size, higher the number of clauses, which will eventually increase the number of iterations to find the satisfiability for 100 instances.