Name: Siqi Zhou

Student Id: 903274

User Name: szhou7

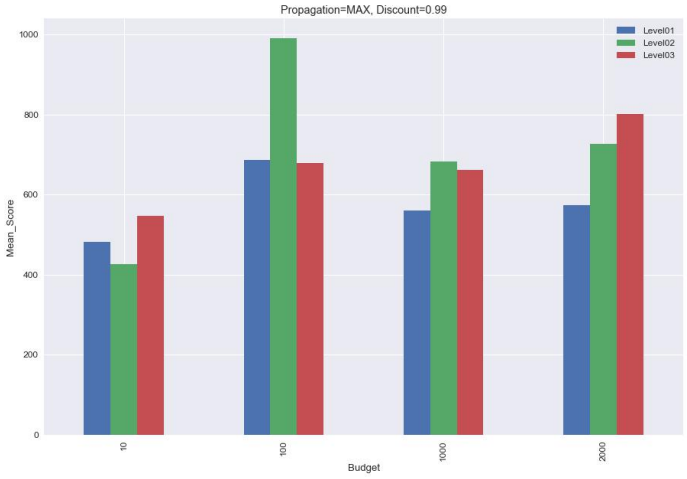
Comp20003 Assignment2 experimentation

Introduction:

The functions were implemented through the pseudo code given in the specification, however, there still exist some differences. For the 'propagateBackScoreToFirstAction', it runs after finishing all searching and takes the 'explored' array which contains all the nodes in the tree to search for the highest score in each direction and store them in an array and pass it to another function called 'find_best_move' to find out which direction has the highest score, and also, if there are more than one highest score, break ties randomly. To do it in this way could be easier when using the average propagation, because average needs to get the reward of all the most bottom nodes, if doing the calculation while still searching for nodes, it is hardly to identify which node is at the most bottom.

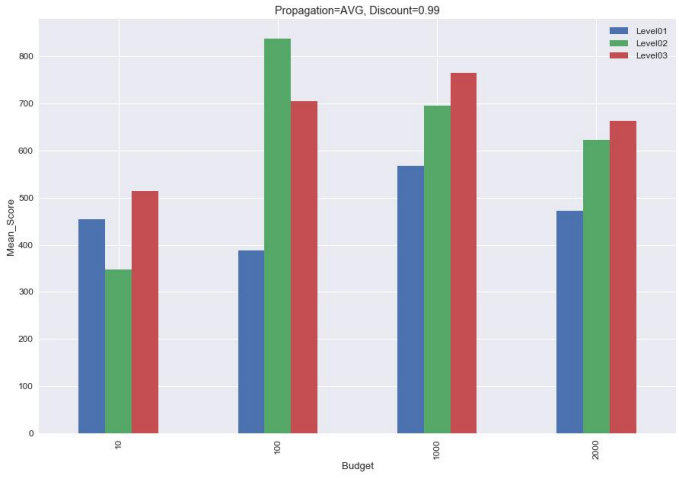
Graphs and tables shows the data collected by changing propagation, budget, level and discount.

Propagation=max, discount=0.99



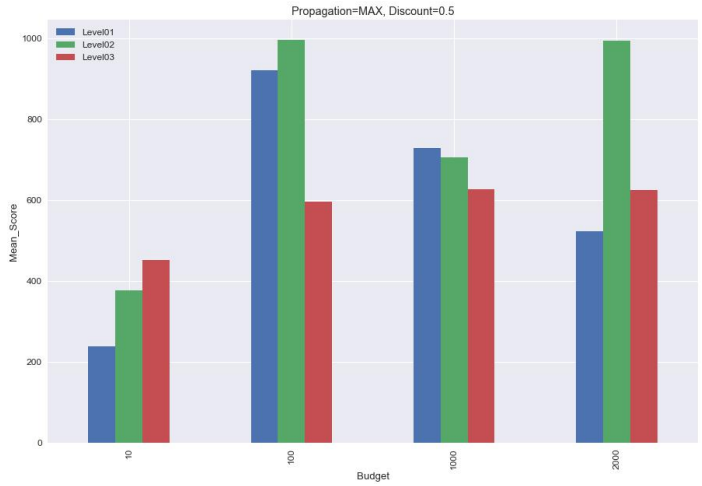
Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	482	10	1005.28089	325	351065.2262
2	427	10	596.2956973		
3	547	10	597.1024133		
1	687	100	11597.97853		
2	991	100	21193.3852		
3	679	100	12754.79024		
1	560	1000	83018.24148		
2	683	1000	73381.26444		
3	662	1000	107768.027		
1	574	2000	122876.7394		
2	728	2000	218060.2453		
3	802	2000	789213.2407		

Propagation=avg, discount=0.99



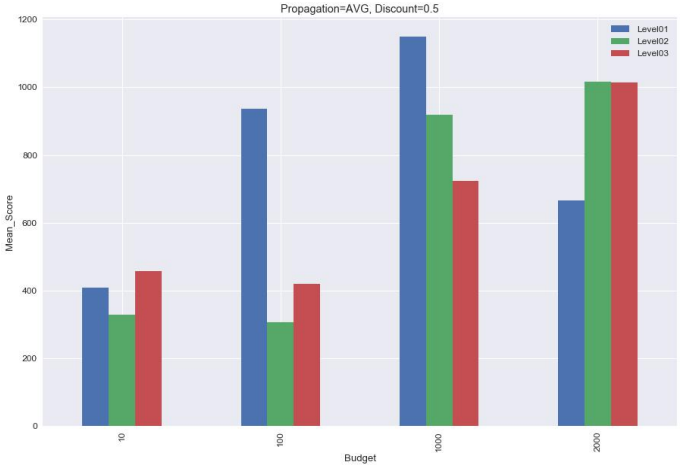
Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	454	10	906.1793517	273	87719.84412
2	347	10	589.7071783		
3	514	10	615.365394		
1	387	100	9047.630001		
2	837	100	15952.77747		
3	706	100	14285.44361		
1	567	1000	67725.30718		
2	695	1000	243499.1466		
3	765	1000	95857.30145		
1	472	2000	91708.60916		
2	623	2000	131636.1607		
3	663	2000	115534.3397		

Propagation=max, discount=0.5



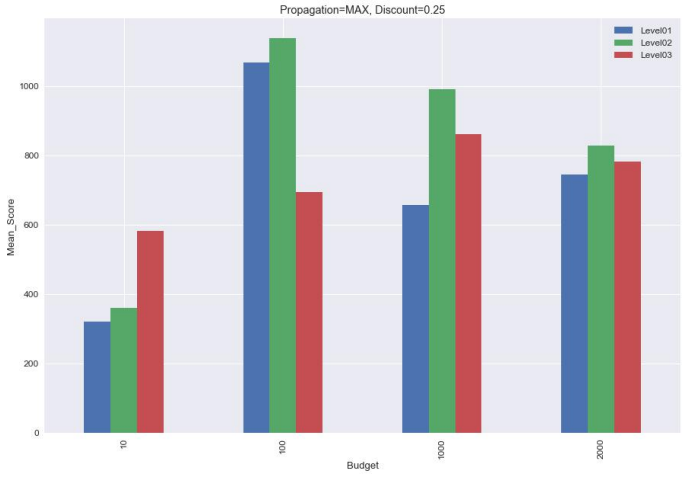
Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	238	10	490.0322817	350	188631.2187
2	378	10	574.078488		
3	452	10	632.352089		
1	921	100	13465.76185		
2	996	100	9423.826544		
3	597	100	10118.79072		
1	729	1000	514914.9312		
2	706	1000	82762.65051		
3	628	1000	104980.8818		
1	524	2000	140451.2682		
2	995	2000	171983.5027		
3	625	2000	92866.93288		

Propagation=avg, discount=0.5



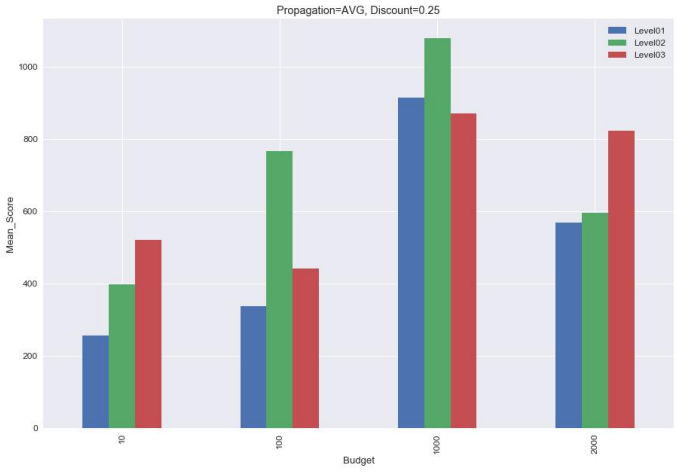
Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	409	10	1033.015408	419	138006.0462
2	329	10	591.2874187		
3	457	10	1305.382437		
1	936	100	25937.74455		
2	306	100	6687.17337		
3	419	100	23010.84876		
1	1149	1000	213315.2442		
2	919	1000	97245.94578		
3	724	1000	66259.83843		
1	666	2000	356973.6473		
2	1017	2000	161338.9764		
3	1015	2000	120203.3486		

Propagation=max, discount=0.25



Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	320	10	516.6940853	420	84340.81632
2	361	10	592.949237		
3	582	10	1175.264409		
1	1067	100	12827.61425		
2	1137	100	17738.65455		
3	695	100	9669.481091		
1	656	1000	108656.2304		
2	990	1000	153051.9681		
3	861	1000	94574.62637		
1	744	2000	169999.2674		
2	828	2000	162301.3553		
3	781	2000	126662.6808		

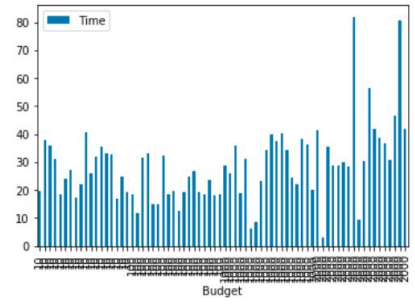
Propagation=avg, discount=0.25



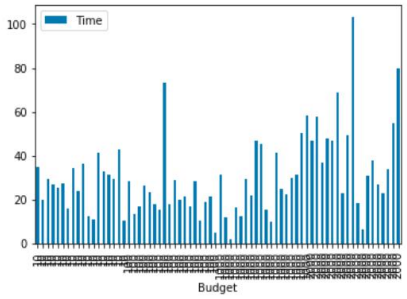
Level	Score_mean	Budget	Expanded/Time_mean	Score_deviation	Expanded/Time_deviation
1	257	10	547.3974583	301	131748.3246
2	398	10	596.765249		
3	521	10	994.2794703		
1	337	100	34896.26299		
2	766	100	12269.83861		
3	441	100	7010.732762		
1	914	1000	181529.3759		
2	1078	1000	164515.8718		
3	870	1000	120394.9717		
1	568	2000	309178.6366		
2	596	2000	146770.7966		
3	823	2000	205933.7076		

Graphs and tables showing the change in time with increasing budget with different discount.

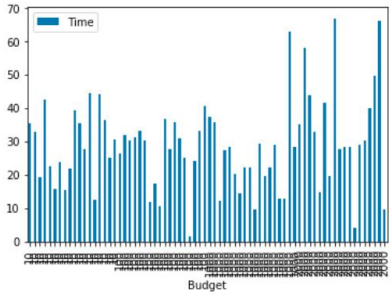
Discount = 0.99



Discount = 0.5



Discount = 0.25



Level	Budget	Propagation	Time_discount=0.99	Time_discount=0.5	Time_discount=0.25
1	10	MAX	19.697011	34.767515	35.432108
1	10	MAX	24.846292	42.702288	25.006413
1	10	MAX	16.726099	29.096087	36.322356
1	100	MAX	24.862258	21.165118	35.756462
1	100	MAX	26.803133	16.782362	30.87962
1	100	MAX	19.425627	28.430109	25.189314
1	1000	MAX	36.092549	2.016602	12.209364
1	1000	MAX	19.04525	16.414281	27.285004
1	1000	MAX	31.278445	12.159187	28.381045
1	2000	MAX	46.721159	33.736128	49.615775
1	2000	MAX	30.668352	22.694996	40.079759
1	2000	MAX	36.53009	26.773937	30.199762
1	10	AVG	40.64719	36.328575	39.360018
1	10	AVG	25.833255	12.442836	35.56623
1	10	AVG	17.186074	34.430081	15.38526
1	100	AVG	17.960603	21.53753	33.114762
1	100	AVG	23.820616	18.660154	24.230037
1	100	AVG	18.444365	10.354075	1.598805
1	1000	AVG	39.757674	15.455717	9.729334
1	1000	AVG	37.559221	9.835839	29.15007
1	1000	AVG	40.20344	41.439075	19.525199
1	2000	AVG	38.852002	38.044041	29.079094
1	2000	AVG	42.054825	30.611907	3.965163
1	2000	AVG	56.421625	6.292476	28.350982

Level	Budget	Propagation	Time_discount=0.99	Time_discount=0.5	Time_discount=0.25
2	10	MAX	37.984372	19.616066	33.010403
2	10	MAX	22.066065	23.99841	21.716749
2	10	MAX	19.326864	10.256018	30.622592
2	100	MAX	18.687325	73.338512	17.394923
2	100	MAX	12.587561	29.023747	36.851344
2	100	MAX	32.176019	15.224295	11.737617
2	1000	MAX	38.130811	31.366932	12.929848
2	1000	MAX	22.066714	29.746488	12.679841
2	1000	MAX	24.542082	22.340773	29.06332
2	2000	MAX	30.211789	18.122154	28.241569
2	2000	MAX	9.552976	103.437162	27.71016
2	2000	MAX	28.676007	47.626566	33.012507
2	10	AVG	18.668213	25.257259	22.536969
2	10	AVG	23.915373	27.504998	15.603853
2	10	AVG	27.387647	15.801762	23.728104
2	100	AVG	11.672801	13.127146	31.737336
2	100	AVG	18.314281	28.298796	26.323019
2	100	AVG	19.685924	17.81175	10.457984
2	1000	AVG	23.290991	46.586838	22.327428
2	1000	AVG	8.709493	21.939447	14.52639
2	1000	AVG	6.10894	29.083424	20.302774
2	2000	AVG	28.233348	22.80709	19.4872
2	2000	AVG	29.9981	68.855304	41.711854
2	2000	AVG	28.811917	47.00525	14.785028

Level	Budget	Propagation	Time_discount=0.99	Time_discount=0.5	Time_discount=0.25
3	10	MAX	35.649963	41.518527	44.499829
3	10	MAX	33.051393	32.967669	12.476817
3	10	MAX	32.594015	31.324763	44.078018
3	100	MAX	31.59068	16.956764	30.433628
3	100	MAX	33.251517	26.267425	31.305553
3	100	MAX	14.824978	23.538772	33.081854
3	1000	MAX	25.997215	11.92033	35.930729
3	1000	MAX	28.672572	31.358347	37.532824
3	1000	MAX	20.180255	58.514215	28.385489
3	2000	MAX	80.507967	54.582199	66.064017
3	2000	MAX	35.440446	36.955763	43.979893
3	2000	MAX	3.082876	57.532489	57.98725
3	10	AVG	32.07744	10.9737	27.623355
3	10	AVG	31.244187	26.74531	42.656311
3	10	AVG	36.058104	29.125231	19.288991
3	100	AVG	14.923168	17.689326	30.180692
3	100	AVG	18.647181	4.746777	40.772041
3	100	AVG	19.279456	19.877713	27.731965
3	1000	AVG	34.410745	24.645017	22.295678
3	1000	AVG	34.1919	45.453151	22.352268
3	1000	AVG	36.407407	50.315576	63.024171
3	2000	AVG	41.336742	46.788128	35.181725
3	2000	AVG	82.040199	49.122632	66.936594
3	2000	AVG	42.035105	79.992906	9.652043

Discussion

Generally, as budget increases, the mean of score also shows an increase trend. However, as indicated by the standard deviations and some unusual behaved bars in the graph, the data is scattered, this is caused by the unpredictable variables, the move of ghosts and the direction chosen by the pacman when there is a tie. When discount is 0.99, the performance of average propagation is overall worse than maximum propagation, but when discount becomes smaller, the difference between the mean score of average and maximum also becomes smaller. Discount will make the rewards of nodes decrease by the power of its depth in the tree, it is used to decrease the proportion of the rewards of more bottom nodes in the tree when counting the total reward for each move, this is because for more bottom nodes in a tree, it is less likely to happen in reality since there exists random variables. Therefore a smaller discount will decrease the difference between the solution of using max and avg, as the nodes closer to the root has higher proportion when counting reward, so that the differences of accumulated reward among most bottom leaf nodes are smaller, and will cause smaller difference when finding the max or calculating the average among these bottom nodes. The expanded nodes/time also shows an increase trend when budget grows, this is because a larger budget will allow the function to expand more nodes.

The total time shows an increase trend as budget increases, because when budget is larger, it will expands more nodes and will needs more time to find the best direction. There are also some bars behave differently, these should also be caused by the random variables, for example, some tests that the pacman successfully get to the next level, it will take longer time to finish a play.