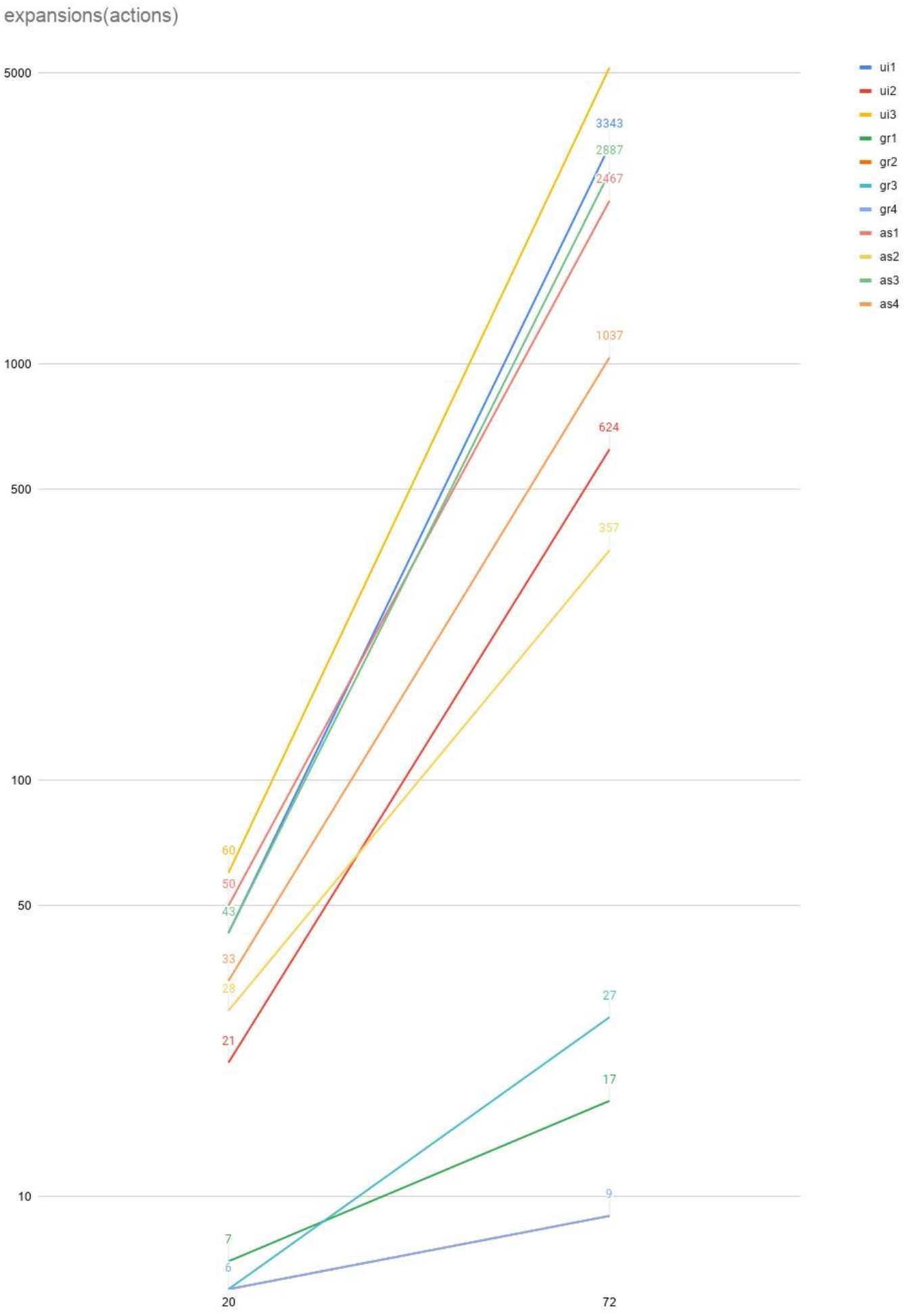
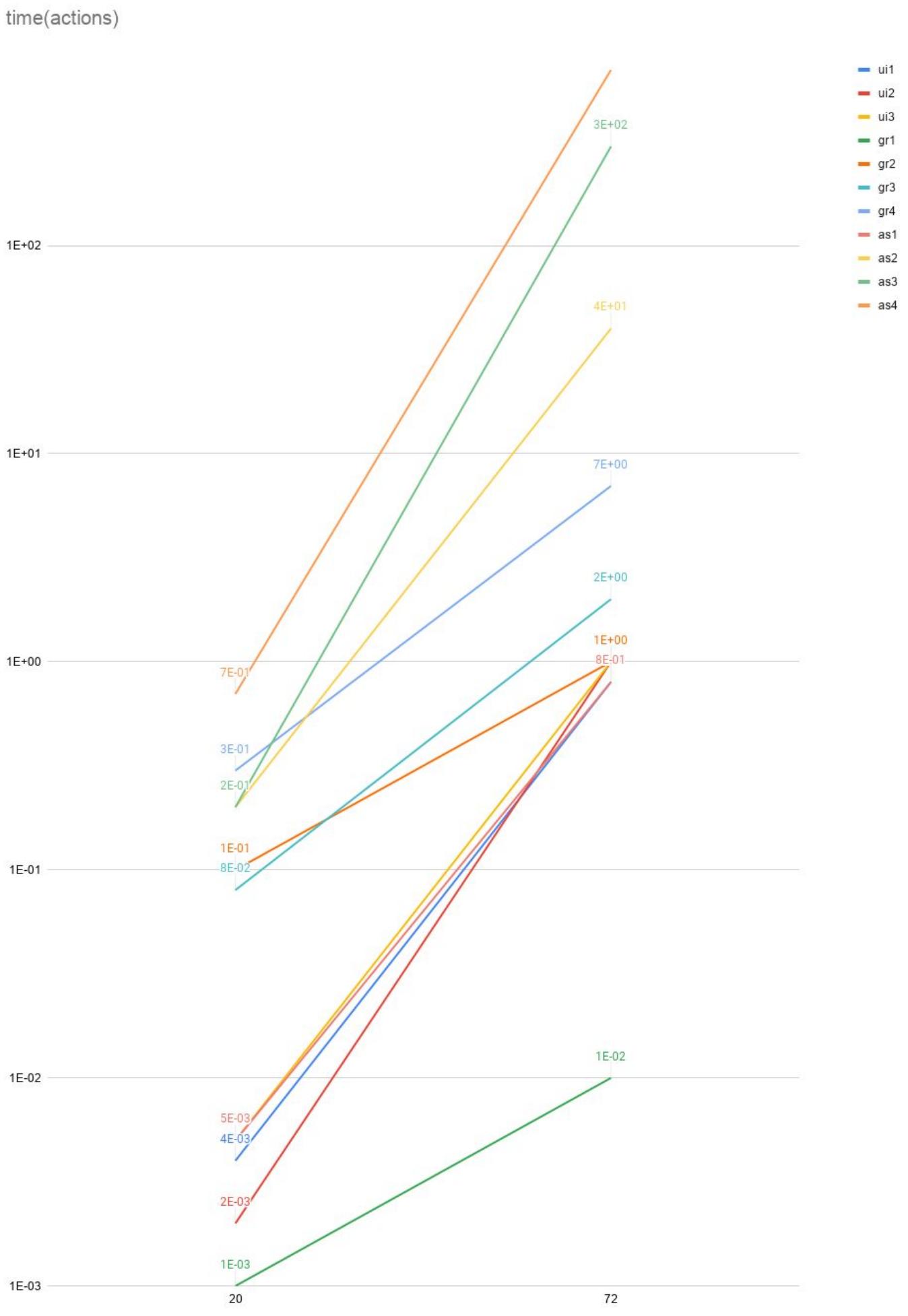
explorative analysis		uninformed search (ui)		
		breadth first	depth first	uniform cost
plan #1 (20 actions)	expansions	43	21	60
	goal tests	56	22	62
	new nodes	178	84	240
	plan length	6	20	6
	elapsed time	4e-03	2e-03	5e-03
plan #2 (72 actions)	expansions	3343	624	5154
	goal tests	4609	625	5156
	new nodes	30503	5602	46618
	plan length	9	619	9
	elapsed time	8e-01	1e+00	1e+00

explorative analysis		greedy best first search (gr)			
		unmet goals	sum level	max level	set level
plan #1 (20 actions)	expansions	7	6	6	6
	goal tests	9	8	8	8
	new nodes	29	28	24	28
	plan length	6	6	6	6
	elapsed time	1e-03	1e-01	8e-02	3e-01
plan #2 (72 actions)	expansions	17	9	27	9
	goal tests	19	11	29	11
	new nodes	170	86	249	84
	plan length	9	9	9	9
	elapsed time	1e-02	1e+00	2e+00	7e+00

explorative analysis		alpha star search (as)			
		unmet goals	sum level	max level	set level
plan #1 (20 actions)	expansions	50	28	43	33
	goal tests	52	30	45	35
	new nodes	206	122	180	138
	plan length	6	6	6	6
	elapsed time	5e-03	2e-01	2e-01	7e-01
plan #2 (72 actions)	expansions	2467	357	2887	1037
	goal tests	2469	359	2889	1039
	new nodes	22522	3426	26594	9605
	plan length	9	9	9	9
	elapsed time	8e-01	4e+01	3e+02	7e+02


expansions(actions)
5000
1000
500
100
50
10
20
72
ui1
ui2
ui3
gr1
gr2
gr3
gr4
as1
as2
as3
as4
3343
2887
2467
1037
624
357
27
17
9
60
50
43
33
28
21
7
6


time(actions)
ui1
ui2
ui3
gr1
gr2
gr3
gr4
as1
as2
as3
as4
1E+02
1E+01
1E+00
1E-01
1E-02
1E-03
20
72
3E+02
4E+01
7E+00
2E+00
1E+00
8E-01
7E-01
3E-01
2E-01
1E-01
8E-02
1E-02
5E-03
4E-03
2E-03
1E-03

for low complexity problems (**plan #1**)

- (**ui**) uninformed search outperforms informed search algorithms as per computing time
- (**gr**) simple heuristic evaluation functions, although being mildly time consuming, optimize node expansion
- (**as**) more advanced search algorithms bring unnecessary complexity for the given planning domain

at a slight increase in complexity (**plan #2**)

- (**ui**) uninformed search expands more nodes than optimal but is optimally fast
- (**gr**) greedy best first search optimizes node expansion at the cost of time consuming evaluations
- (**as**) complex search algorithms still are not worth the implementation

for more complex action spaces (**plan #3 and #4**)

- (**ui**) i avoided depth first search because it returned nonoptimal plans in previous runs
- (**gr**) i selected heuristic evaluation functions for the best performance among the other metrics
- (**as**) i selected heuristic evaluation functions for the best performance among the other metrics

efficient analysis		uninformed search (ui)		greedy best first search (gr)		alpha star search (as)	
		breadth first	uniform cost	unmet goals	max level	unmet goals	sum level
plan #3 (88 actions)	expansions	14663	18510	25	21	7388	369
	goal tests	18098	18512	27	23	7390	371
	new nodes	129625	161936	230	195	65711	3403
	plan length	12	12	15	13	12	12
	elapsed time	4e+00	5e+00	1e-02	4e+00	3e+00	9e+01
plan #2 (104 actions)	expansions	99736	113339	29	56	34330	1208
	goal tests	114953	113341	31	58	34332	1210
	new nodes	944130	1066413	280	580	328509	12210
	plan length	14	14	18	17	14	15
	elapsed time	4e+01	4e+01	3e-02	1e+01	2e+01	5e+02

in conclusion and as a default approach for future classical planning problems

- appropriate algorithms for restricted planning domains are uninformed or based on shallow heuristics
- planning in very large domains should optimize node expansion more than and before other metrics
- breadth first and uniform cost uninformed search algorithms are guaranteed to find optimal solutions