Experimental results

YILING ZENG, Nankai University, China
CHUNYAO SONG, Nankai University, China
TINGJIAN GE, University of Massachusetts Lowell, USA

1 RUNNING TIME

1.1 Sampling time

The total running time includes the sampling time on the initial graph and the graph analysis time on the sampled graph. Table 1 shows the sampling time of three methods with different values of p. It can be seen that when the size of the datasets grows exponentially, the sampling time of ABM is almost unchanged, and ADR can achieve nearly linear growth. While the time cost of UDS is too large that it cannot finish sampling on com-LiveJournal dataset within 10 times of ADR's sampling time, so we give up sampling on com-LiveJournal by UDS. In contrast, ABM can complete the corresponding sampling tasks within 500s. The above results fully demonstrate the superiority of ADR and ABM in their sampling abilities, indicating that these two techniques can achieve fast sampling with limited resources (a personal desktop).

Table 1. Sampling time (sec)

p	ca-GrQc			ca-HepPh			email-Enron			com-LiveJournal		
	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	15.212	14.861	0.257	268.972	275.619	1.084	6879.807	1885.879	2.645		3626.847	357.569
0.8	15.207	14.925	0.250	271.939	294.612	1.250	55965.467	1829.306	2.537		3312.434	397.061
0.7	15.426	14.965	0.258	302.764	315.623	1.423	160353.568	1956.268	2.889		3223.822	365.409
0.6	16.599	14.789	0.279	890.657	318.554	1.543	231575.210	1956.607	3.101		2631.308	459.907
0.5	19.217	14.934	0.257	3054.891	292.064	1.642	296826.905	1822.863	3.674		2426.456	293.872
0.4	27.516	14.510	0.297	5269.170	307.329	1.758	422570.755	1873.002	3.639		2043.653	323.950
0.3	66.699	14.510	0.283	8057.549	300.614	1.880	497718.257	1832.783	3.837		1776.021	314.252
0.2	179.200	13.895	0.359	11284.950	274.653	2.039	562725.773	1819.197	4.058		1419.634	343.632
0.1	365.766	13.246	0.349	15773.001	241.629	2.399	604679.461	1857.304	4.161		1096.614	330.989

1.2 Graph analysis time

Next, we show the graph analysis time of all graph analysis tasks on sampled graphs of all datasets in Tables 2-7.

Tables 2-3 are graph analysis time of ca-GrQc which are divided into two parts by the time complexity of graph analysis tasks.

Authors' addresses: Yiling Zeng, yilingzeng@mail.nankai.edu.cn, Nankai University, Tianjin, China; Chunyao Song, chunyao.song@nankai.edu.cn, Nankai University, Tianjin, China; Tingjian Ge, ge@cs.uml.edu, University of Massachusetts Lowell, USA.

Table 2. Graph analysis time on sampled graph on ca-GrQc *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	1.016			0.075			0.202		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	0.954	0.961	0.870	0.100	0.046	0.033	0.204	0.188	0.178
0.8	0.847	0.995	0.841	0.103	0.038	0.037	0.288	0.355	0.203
0.7	0.798	1.004	0.969	0.092	0.032	0.037	0.531	0.223	0.165
0.6	0.659	0.929	0.889	0.101	0.025	0.025	0.260	0.164	0.142
0.5	0.475	0.829	0.699	0.076	0.021	0.018	0.332	0.113	0.088
0.4	0.388	0.721	0.668	0.090	0.049	0.047	0.089	0.081	0.071
0.3	0.315	0.619	0.584	0.031	0.038	0.027	0.116	0.075	0.058
0.2	0.248	0.496	0.476	0.034	0.009	0.034	0.308	0.057	0.052
0.1	0.107	0.237	0.177	0.034	0.005	0.004	0.039	0.024	0.015

Table 3. Graph analysis time on sampled graph on ca-GrQc *II* (sec)

	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	321.68			74.182			110.466			141.429		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	311.494	301.712	201.694	55.883	65.899	61.365	61.160	77.904	75.910	109.176	102.577	134.857
0.8	271.387	159.566	149.595	40.568	65.881	57.396	45.701	78.630	77.709	73.941	131.086	141.480
0.7	245.125	120.533	120.536	27.017	52.776	46.042	33.155	71.087	62.356	38.712	84.506	83.967
0.6	154.945	109.495	104.404	22.709	47.032	41.408	25.178	55.414	47.921	38.732	94.016	79.196
0.5	110.855	91.331	71.134	19.027	39.864	24.273	21.561	50.586	32.131	28.123	66.137	50.235
0.4	92.728	71.282	67.180	13.789	25.118	20.552	16.017	33.526	27.559	19.032	42.300	36.052
0.3	55.562	58.061	53.955	7.296	11.773	9.719	8.745	18.729	15.441	9.369	21.520	19.981
0.2	43.374	37.630	30.595	2.707	3.420	2.345	3.456	7.431	5.774	4.054	5.120	3.960
0.1	20.209	10.261	5.203	0.695	0.329	0.172	0.821	1.605	0.664	1.239	1.080	0.486

Tables 4-5 are graph analysis time of ca-HepPh and Tables 6-7 are for email-Enron. The "T" lines represent processing time on the initial graph.

According to the tables, we can see that the three sampling methods can directly reduce the evaluation time of graph analysis tasks in most cases. Due to the fundamental process of the sampled graphs, the time has risen temporarily in some cases. Also, the performances of the three methods are not consistent on different graph analysis tasks. However, with the increase in the datasets' size and the time complexity, the reduction in graph analysis time becomes more and more obvious.

Table 4. Graph analysis time on sampled graph on ca-HepPh *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	4.707			0.620			6.208		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	4.360	4.064	3.703	0.635	0.314	0.281	6.412	5.585	4.562
0.8	3.542	3.768	3.391	0.552	0.278	0.240	3.596	4.192	3.396
0.7	3.438	3.734	3.314	0.462	0.249	0.230	3.356	3.549	2.629
0.6	2.956	3.365	2.919	0.377	0.222	0.188	3.626	3.306	2.269
0.5	2.882	3.040	3.031	0.332	0.177	0.154	3.172	1.887	1.498
0.4	2.411	2.542	2.417	0.310	0.150	0.135	3.713	1.266	1.305
0.3	2.034	2.314	2.112	0.291	0.112	0.104	2.417	0.890	0.717
0.2	1.471	1.692	1.601	0.291	0.078	0.073	1.485	0.431	0.381
0.1	0.813	0.976	0.929	0.228	0.042	0.038	0.652	0.168	0.140

Table 5. Graph analysis time on sampled graph on ca-HepPh *II* (sec)

	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	993.651			1022.666			1065.888			734.352		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	938.605	1013.316	899.838	1155.423	1156.014	987.895	1037.414	1110.107	1002.609	934.770	874.975	781.663
0.8	886.301	965.361	890.256	802.823	991.358	844.841	874.987	1006.259	877.055	584.928	834.084	614.790
0.7	770.941	866.165	800.407	670.931	904.971	721.090	717.743	959.308	826.975	486.002	759.890	579.873
0.6	533.807	793.444	742.594	381.834	812.281	621.121	369.374	873.451	667.687	249.428	661.019	492.841
0.5	397.218	758.245	621.899	229.009	657.288	415.906	202.154	758.655	500.095	190.641	583.667	337.413
0.4	300.244	685.057	620.611	140.064	510.242	375.628	115.761	571.248	463.112	119.293	433.284	296.542
0.3	207.503	621.853	552.080	65.076	357.544	243.166	57.808	433.071	319.263	67.160	296.212	190.589
0.2	124.331	504.522	446.152	23.411	192.074	128.220	22.942	252.532	180.587	39.812	165.116	101.336
0.1	59.754	323.841	266.657	5.745	60.097	35.943	7.379	99.985	63.255	14.123	52.145	29.508

Table 6. Graph analysis time on sampled graph on email-Enron *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	10.407			1.152			10.489		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	10.216	8.299	7.146	1.171	0.655	0.547	15.012	14.749	8.206
0.8	8.562	7.081	6.207	1.080	0.569	0.489	12.295	10.273	6.324
0.7	5.664	6.281	5.766	0.970	0.496	0.404	10.179	7.102	4.941
0.6	5.315	5.653	5.234	0.922	0.436	0.370	8.946	6.242	4.061
0.5	4.402	5.406	4.818	0.857	0.373	0.279	7.507	4.635	2.999
0.4	3.510	4.071	3.726	0.840	0.292	0.244	5.917	4.333	2.310
0.3	2.956	3.497	3.182	0.889	0.232	0.198	4.676	1.879	1.555
0.2	2.370	2.565	2.369	0.841	0.158	0.138	2.338	1.182	1.254
0.1	3.211	1.501	1.318	0.865	0.088	0.063	2.350	0.574	0.416

Table 7. Graph analysis time on sampled graph on email-Enron *II* (sec)

	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	3302.558			13716.537			20290.111			16568.052		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	2643.240	3533.481	2804.757	11828.763	14585.398	11652.593	13299.943	22405.572	14688.822	12015.097	16560.067	12049.356
0.8	1661.849	3302.317	2482.619	5686.872	13568.386	8126.157	5777.243	19370.400	10636.012	5847.883	15570.875	9048.955
0.7	1219.096	3182.403	2261.632	3168.565	11509.695	5942.541	2906.628	16955.713	7881.167	3913.915	12975.282	7856.792
0.6	846.712	2852.295	1989.631	2002.306	10373.906	4387.991	1736.412	14041.168	5851.309	2052.895	11640.793	4577.721
0.5	682.331	2657.078	1688.226	1212.559	7811.894	3158.780	1030.596	11397.758	3816.374	1300.628	10465.631	3729.803
0.4	459.535	2171.175	1489.478	694.661	6202.025	2688.127	588.475	8792.200	3110.560	729.241	9054.873	2469.503
0.3	314.045	1803.180	1251.544	365.843	4061.638	1533.525	310.089	6071.446	2130.359	367.417	4305.307	1661.734
0.2	219.201	1542.456	1154.076	121.820	2211.570	745.347	101.140	3332.773	1103.271	124.541	2387.980	818.730
0.1	122.399	801.788	499.911	34.253	637.284	196.764	36.333	890.703	274.340	33.591	1003.209	344.473

1.3 Total time

Next, let us examine the whole processing time (sampling time plus graph analysis time on sampled graph) for different graph analysis tasks on all datasets. The results are displayed in Tables 8-13, where the "T" lines show the processing time on initial graph. Also, we display these tasks by their time complexity.

For Tables 8, 10 and 12, since the time complexity of these three graph analysis tasks, Top-k query, Vertex degree and Clustering coefficient, is low, the whole processing time using sampling methods do not present significant advantages comparing to performing the graph analysis tasks directly on the initial graph. However, we can still find ADR and ABM surpass UDS a lot especially when we need a small sample size. Moreover, in practical scenarios, the sampled graph can be reused after being generated and the time-saving effect cannot be judged based on only one time of use. As a conclusion, ADR and ABM can further reduce the processing time for graph analysis tasks with low time complexities.

Tables 9, 11 and 13 show the results for the rest four graph analysis tasks with relatively high time complexities. Again, we can see ADR and ABM perform much better than UDS. Moreover, ADR and ABM exhibit great efficiency comparing with performing graph analysis tasks directly on original graphs, especially when sample size is small.

Table 8. Total processing time on ca-GrQc *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	1.016			0.075			0.202		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	16.165	15.822	1.127	15.312	14.907	0.290	15.415	15.049	0.435
0.8	16.054	15.920	1.091	15.309	14.962	0.287	15.495	15.280	0.453
0.7	16.224	15.968	1.227	15.518	14.996	0.295	15.957	15.187	0.423
0.6	17.258	15.718	1.168	16.700	14.814	0.304	16.858	14.953	0.421
0.5	19.693	15.763	0.956	19.293	14.955	0.276	19.550	15.046	0.345
0.4	27.905	15.231	0.965	27.607	14.559	0.344	27.605	14.591	0.368
0.3	67.014	15.129	0.867	66.730	14.548	0.310	66.814	14.585	0.341
0.2	179.448	14.391	0.835	179.234	13.904	0.393	179.508	13.952	0.411
0.1	365.873	13.483	0.526	365.801	13.252	0.353	365.805	13.271	0.365

Table 9. Total processing time on ca-GrQc *II* (sec)

	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	321.68			74.182			110.466			141.429		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	326.706	316.573	201.951	71.094	80.760	61.623	76.372	92.765	76.167	124.388	117.438	135.114
0.8	286.594	174.490	149.845	55.775	80.806	57.646	60.908	93.555	77.958	89.148	146.011	141.730
0.7	260.551	135.497	120.794	42.443	67.740	46.300	48.582	86.052	62.613	54.138	99.471	84.225
0.6	171.544	124.284	104.683	39.307	61.821	41.687	41.777	70.203	48.200	55.331	108.805	79.475
0.5	130.072	106.264	71.391	38.245	54.798	24.530	40.779	65.520	32.388	47.340	81.070	50.492
0.4	120.244	85.792	67.478	41.305	39.628	20.849	43.533	48.036	27.856	46.549	56.810	36.350
0.3	122.261	72.571	54.238	73.995	26.283	10.002	75.444	33.239	15.725	76.067	36.030	20.264
0.2	222.574	51.525	30.954	181.907	17.315	2.704	182.656	21.326	6.133	183.254	19.015	4.319
0.1	385.975	23.507	5.552	366.462	13.575	0.522	366.588	14.851	1.014	367.006	14.326	0.836

Table 10. Total processing time on ca-HepPh *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	4.707			0.620			6.208		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	273.333	279.683	4.788	269.607	275.933	1.365	275.384	281.204	5.646
0.8	275.481	298.380	4.641	272.491	294.890	1.491	275.536	298.804	4.646
0.7	306.201	319.357	4.738	303.225	315.872	1.654	306.119	319.172	4.052
0.6	893.613	321.920	4.462	891.033	318.776	1.731	894.283	321.860	3.812
0.5	3057.774	295.105	4.673	3055.223	292.241	1.796	3058.063	293.951	3.140
0.4	5271.582	309.871	4.175	5269.481	307.478	1.893	5272.884	308.595	3.063
0.3	8059.583	302.929	3.993	8057.840	300.726	1.984	8059.966	301.504	2.598
0.2	11286.421	276.345	3.639	11285.241	274.732	2.111	11286.435	275.084	2.420
0.1	15773.814	242.605	3.329	15773.229	241.671	2.437	15773.653	241.798	2.539

Table 11. Total processing time on ca-HepPh *II* (sec)

	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	993.651			1022.666			1065.888			734.352		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	1207.577	1288.935	900.923	1424.395	1431.633	988.979	1306.387	1385.727	1003.694	1203.742	1150.594	782.747
0.8	1158.240	1259.974	891.507	1074.762	1285.971	846.092	1146.926	1300.872	878.306	856.867	1128.696	616.040
0.7	1073.705	1181.788	801.830	973.694	1220.594	722.513	1020.507	1274.931	828.398	788.765	1075.514	581.296
0.6	1424.463	1111.998	744.138	1272.491	1130.836	622.664	1260.031	1192.005	669.231	1140.085	979.574	494.384
0.5	3452.109	1050.310	623.541	3283.900	949.353	417.548	3257.045	1050.719	501.738	3245.533	875.731	339.055
0.4	5569.415	992.386	622.369	5409.234	817.570	377.386	5384.931	878.576	464.870	5388.463	740.613	298.300
0.3	8265.053	922.467	553.960	8122.625	658.158	245.046	8115.357	733.685	321.144	8124.710	596.826	192.469
0.2	11409.281	779.176	448.190	11308.361	466.728	130.259	11307.893	527.185	182.626	11324.762	439.769	103.374
0.1	15832.755	565.470	269.056	15778.746	301.726	38.342	15780.380	341.614	65.654	15787.124	293.774	31.908

Table 12. Total processing time on email-Enron *I* (sec)

	Top-k			Vertex degree			Clustering coefficient		
T	10.407			1.152			10.489		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	6890.023	1894.179	9.792	6880.978	1886.535	3.193	6894.819	1900.629	10.851
0.8	55974.029	1836.387	8.743	55966.547	1829.875	3.026	55977.763	1839.579	8.860
0.7	160359.232	1962.549	8.655	160354.539	1956.764	3.292	160363.748	1963.371	7.830
0.6	231580.526	1962.260	8.335	231576.132	1957.043	3.472	231584.156	1962.849	7.162
0.5	296831.307	1828.269	8.492	296827.762	1823.236	3.953	296834.412	1827.499	6.673
0.4	422574.265	1877.073	7.365	422571.595	1873.294	3.883	422576.672	1877.335	5.949
0.3	497721.214	1836.280	7.018	497719.146	1833.015	4.035	497722.934	1834.662	5.392
0.2	562728.144	1821.762	6.427	562726.615	1819.354	4.196	562728.112	1820.378	5.312
0.1	604682.672	1858.806	5.479	604680.326	1857.393	4.224	604681.811	1857.878	4.577

Table 13. Total processing time on email-Enron *II* (sec)

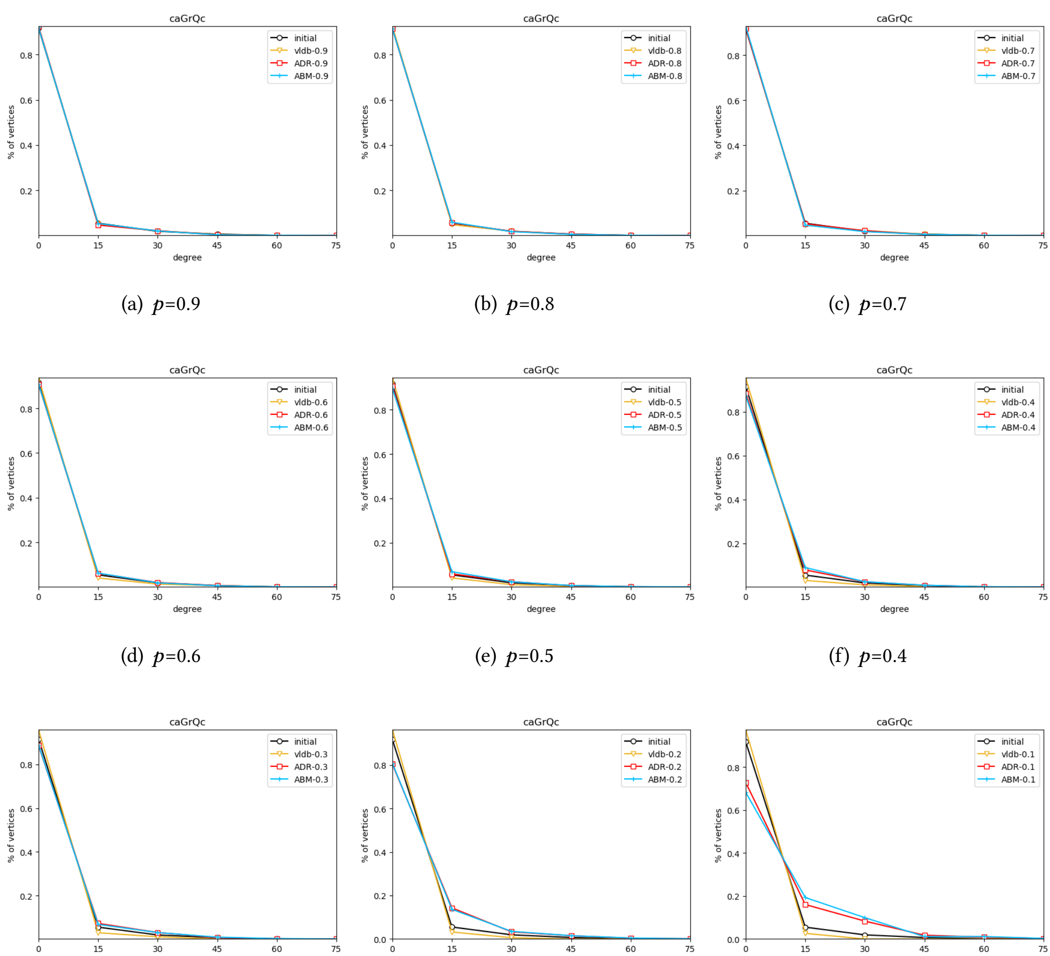
	Link prediction			SP distance			Betweenness centrality			Hop-plot		
T	3302.558			13716.537			20290.111			16568.052		
p	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	9523.047	5419.360	2807.403	18708.570	16471.277	11655.238	20179.750	24291.452	14691.468	18894.903	18445.947	12052.001
0.8	57627.317	5131.623	2485.156	61652.339	15397.692	8128.694	61742.710	21199.706	10638.549	61813.350	17400.181	9051.492
0.7	161572.664	5138.671	2264.520	163522.133	13465.964	5945.430	163260.196	18911.981	7884.056	164267.484	14931.550	7859.680
0.6	232421.923	4808.902	1992.733	233577.516	12330.513	4391.093	233311.623	15997.775	5854.410	233628.105	13597.400	4580.822
0.5	297509.236	4479.941	1691.900	298039.464	9634.757	3162.454	297857.501	13220.622	3820.048	298127.532	12288.494	3733.477
0.4	423030.290	4044.177	1493.117	423265.416	8075.027	2691.766	423159.230	10665.202	3114.199	423299.996	10927.875	2473.142
0.3	498032.302	3635.963	1255.381	498084.101	5894.421	1537.362	498028.346	7904.229	2134.195	498085.674	6138.090	1665.571
0.2	562944.975	3361.653	1158.134	562847.593	4030.766	749.405	562826.913	5151.970	1107.329	562850.314	4207.176	822.788
0.1	604801.860	2659.092	504.072	604713.714	2494.588	200.925	604715.795	2748.007	278.501	604713.052	2860.513	348.634

2 SAMPLING QUALITY

Next we focus on the sampling quality of ADR and ABM with respect to UDS on different graph analysis tasks.

2.1 Vertex degree

Figures 1-3 show the vertex degree distribution on different datasets. Since email-Enron dataset has a wide degree range, the vertex degrees larger than 300 are aggregated as 300. For ca-GrQc, we can see that the results of ADR and ABM are very close to the initial vertex degree distributions when p is larger from Figure 1(a)-1(e). When p is smaller, there is a slight difference between the ADR, ABM and the inital graph from Figure 1(f)-1(i). UDS performs a little worse than ADR and ABM on the whole. As for other datasets, the performances are similar to ca-GrQc.

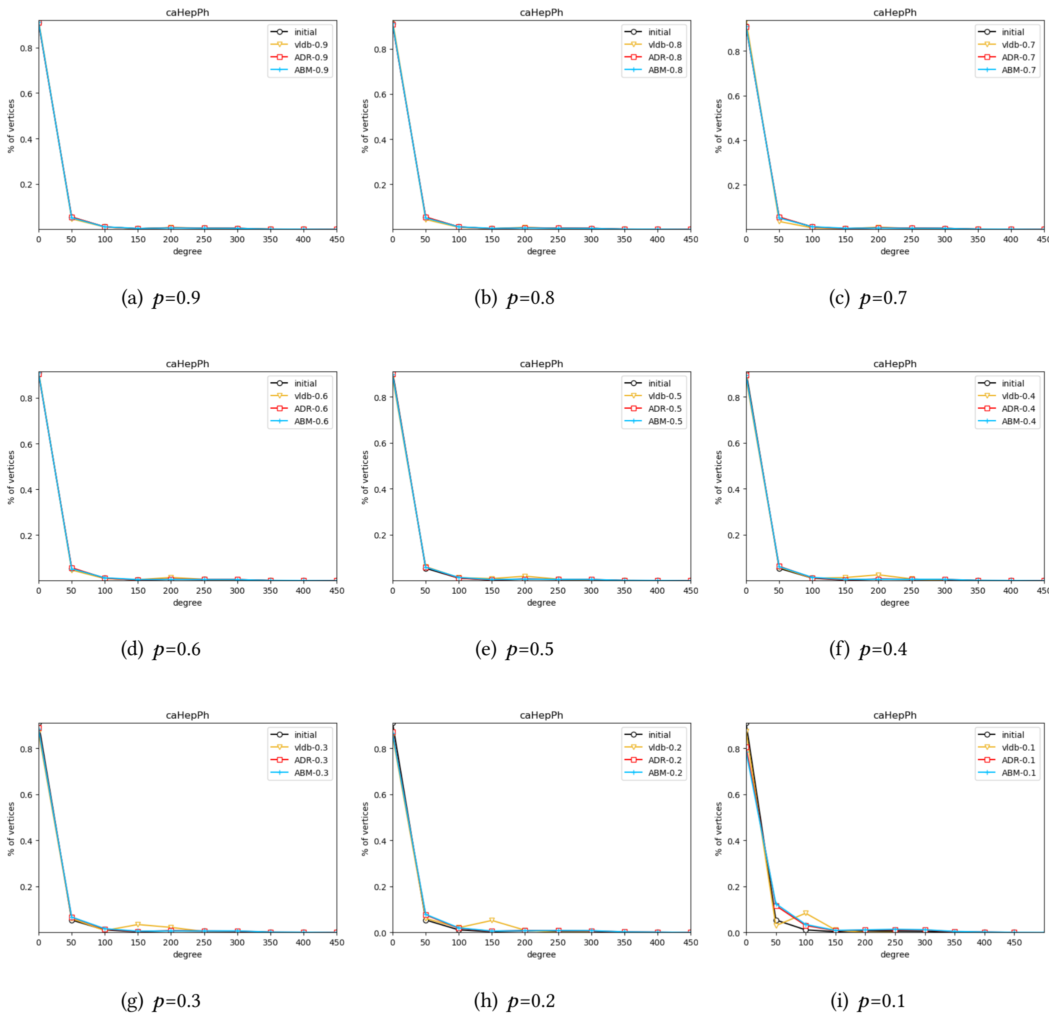


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 1. Vertex degree of ca-GrQc

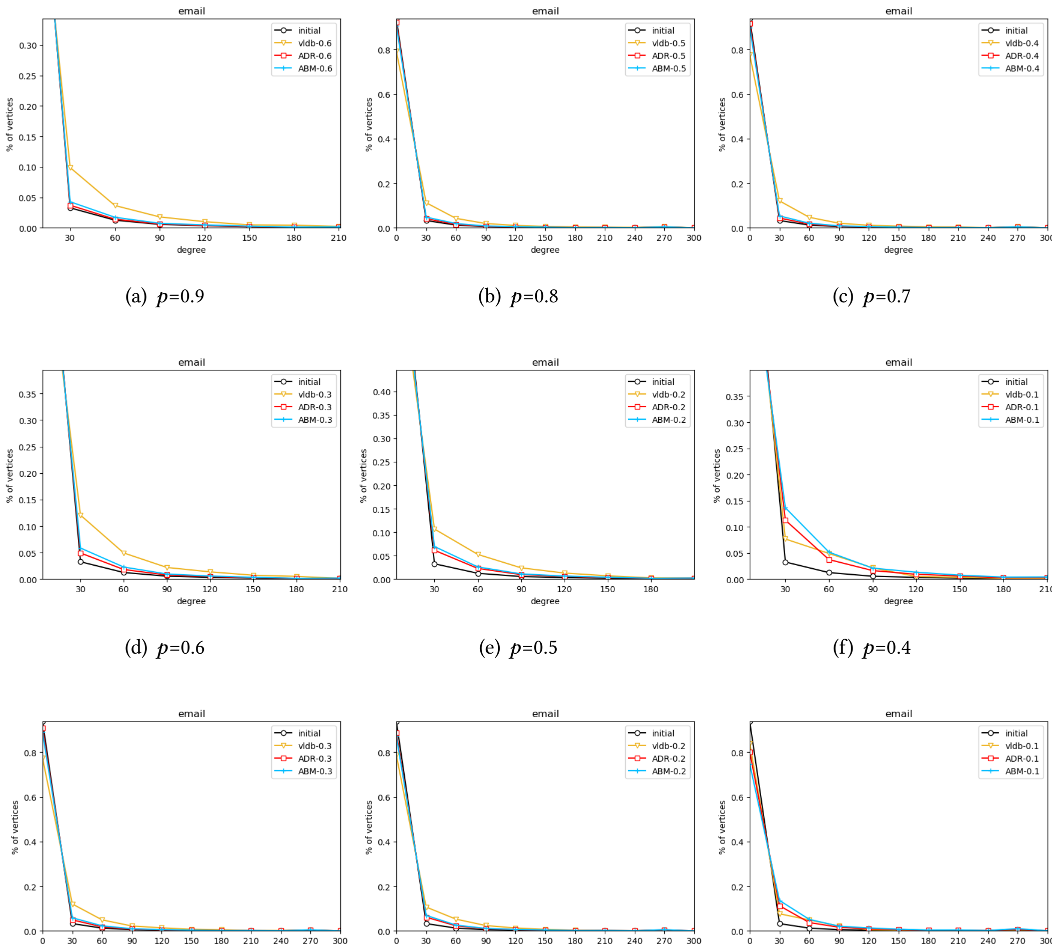


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 2. Vertex degree of ca-HepPh



(a) p=0.9 (b) p=0.8 (c) p=0.7

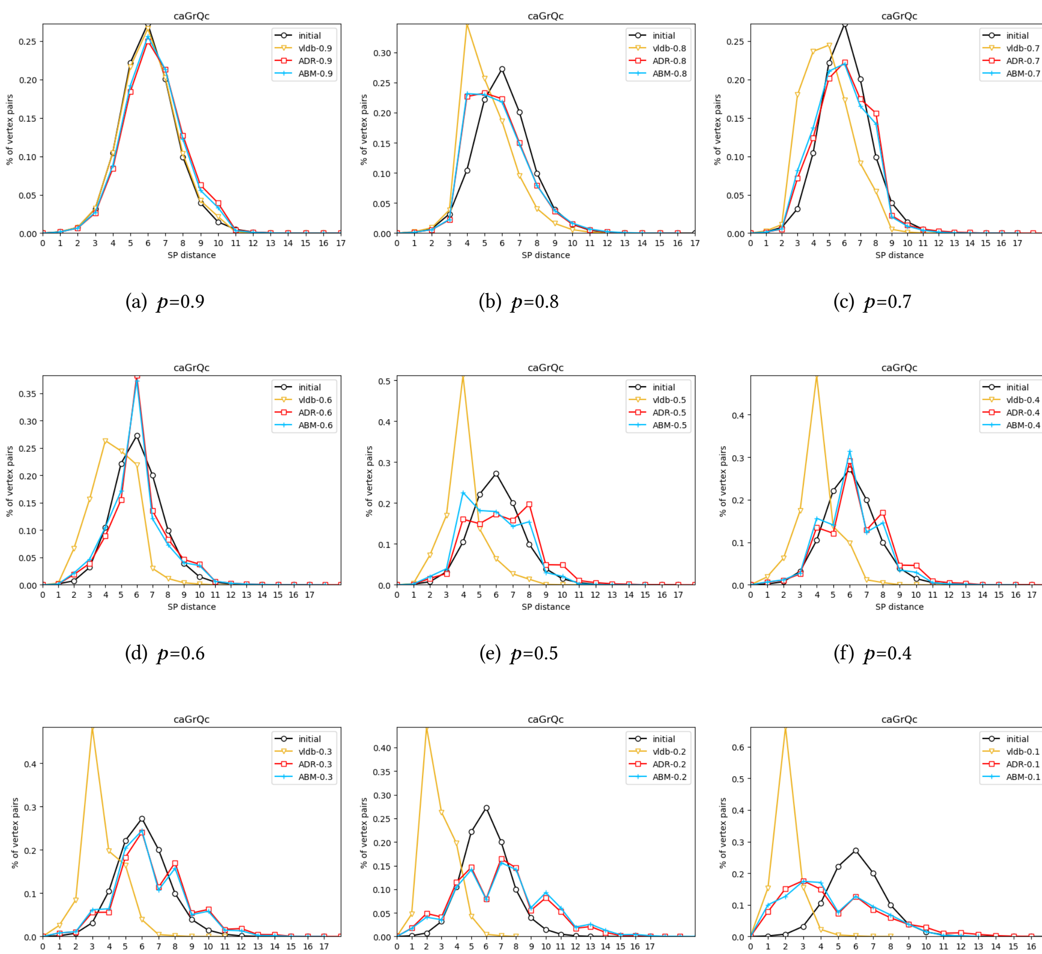
(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 3. Vertex degree of email-Enron

2.2 Shortest-path distance

Figures 4-6 illustrate the shortest-path distance distribution. From Figures 4(a)-4(c), we can see that the results of three methods are close to the initial graph. And for other values of p which are shown in Figures 4(d)-4(i), the results of ADR and ABM are consistent with the trend of the initial graph, but there are numerical differences. UDS is deviated from the original curve when p is smaller. It can be seen that ADR and ABM maintain the shortest-path distance better, while UDS has lost this feature when p is smaller.

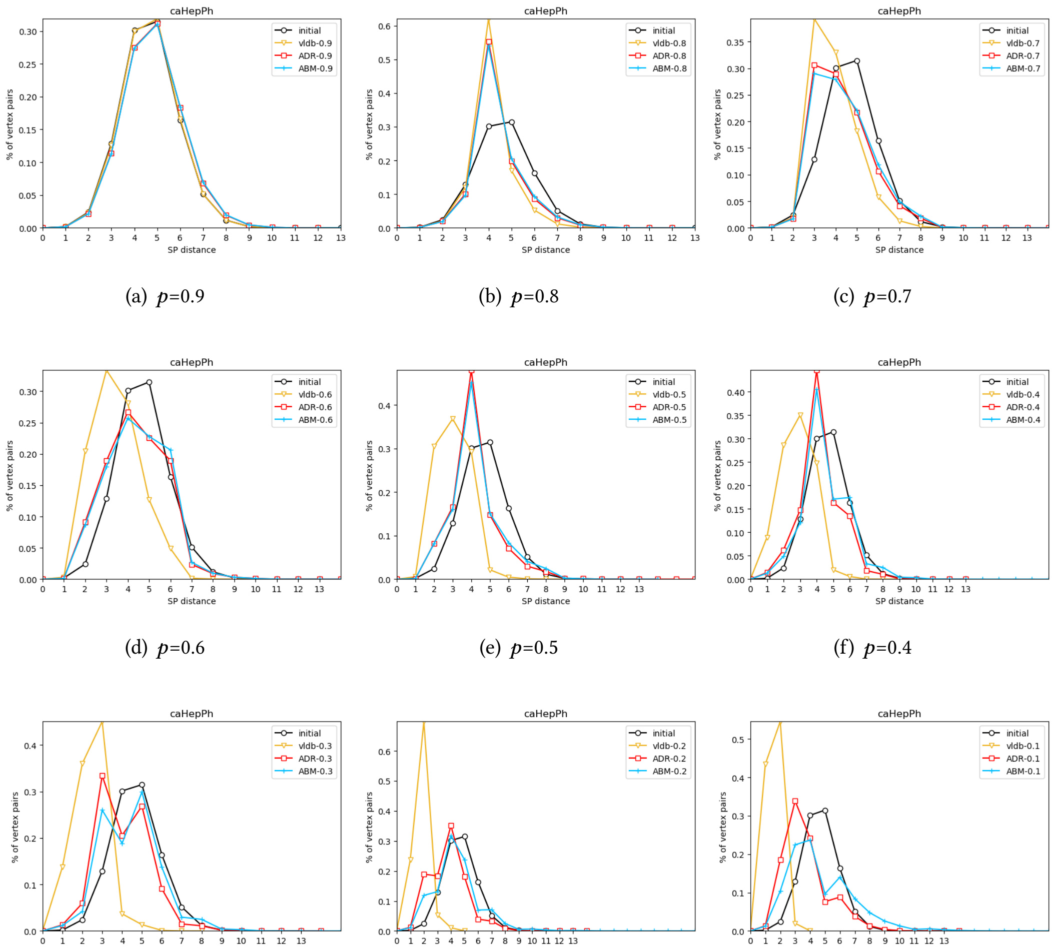


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 4. Shortest path distance of ca-GrQc

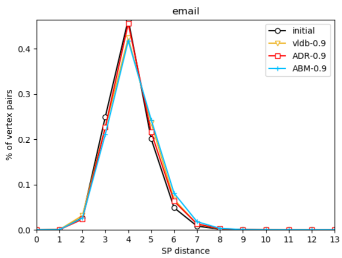


(a) p=0.9 (b) p=0.8 (c) p=0.7

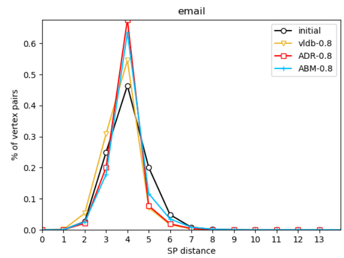
(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

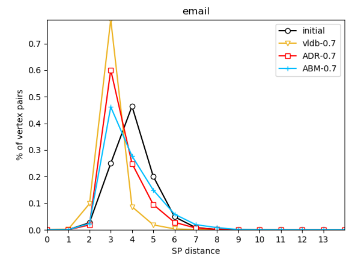
Fig. 5. Shortest path distance of ca-HepPh



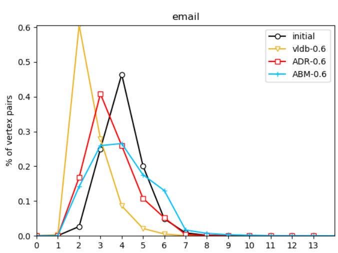
(a) p=0.9



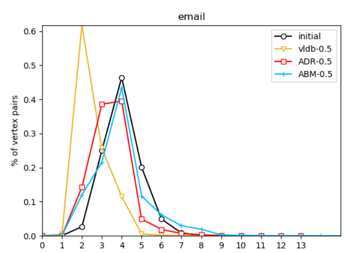
(b) p=0.8



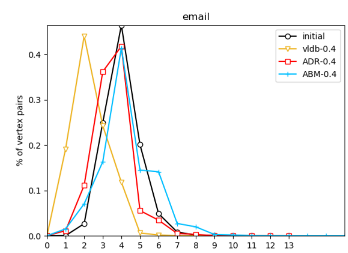
(c) p=0.7



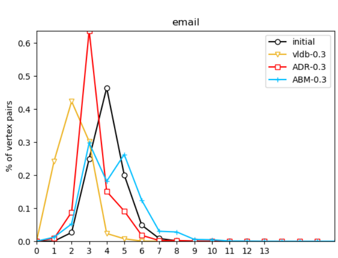
(d) p=0.6



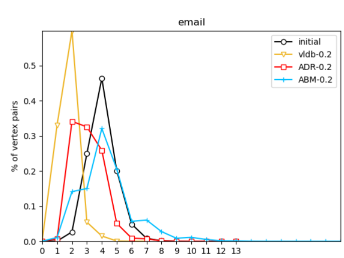
(e) p=0.5



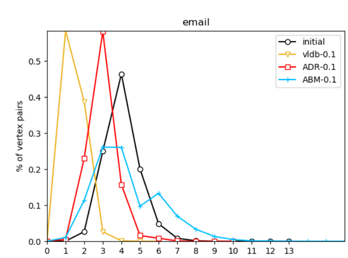
(f) p=0.4



(g) p=0.3



(h) p=0.2

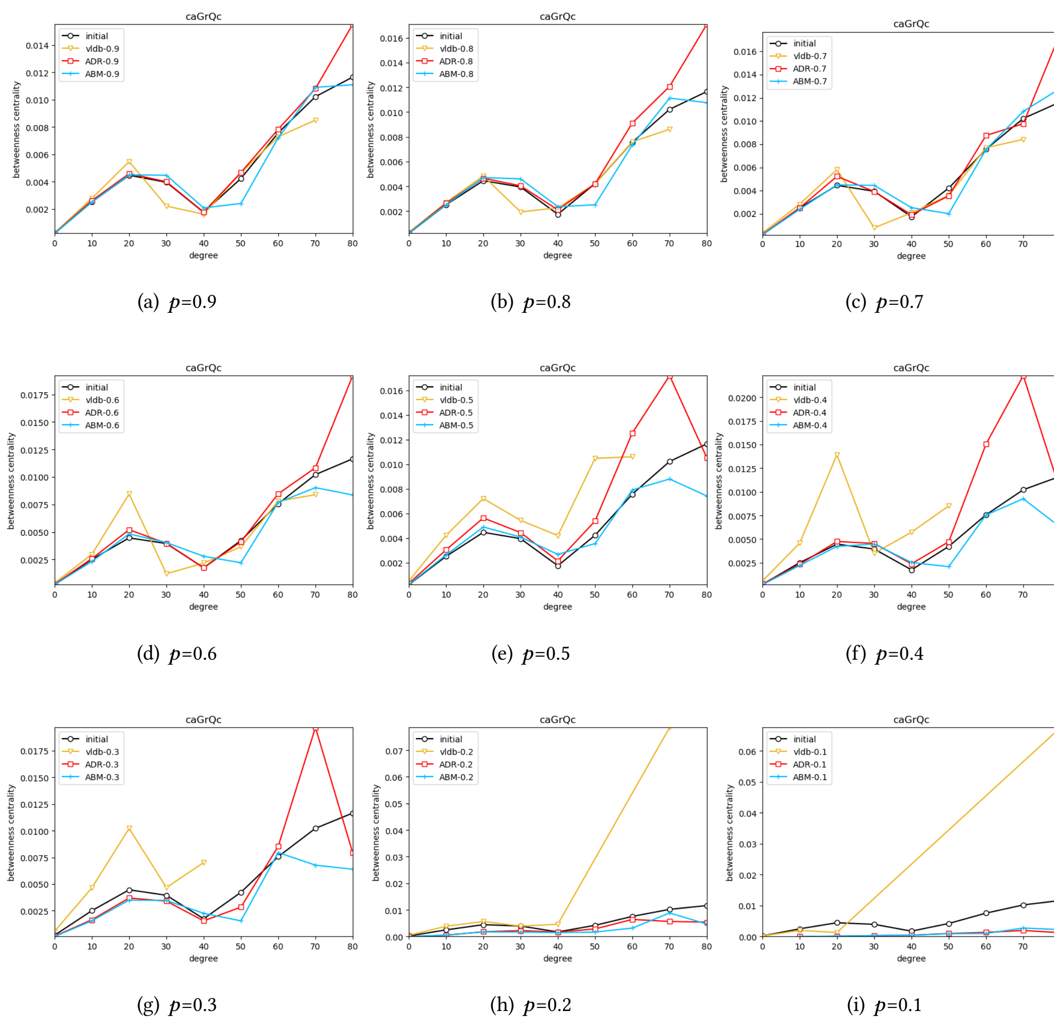


(i) p=0.1

Fig. 6. Shortest path distance of email-Enron

2.3 Betweenness centrality

Figures 7-9 show the betweenness centrality versus the vertex degree. We can see that ADR and ABM measure the betweenness centrality of vertices with lower degrees very accurately in all cases, but the measure of vertices with higher degrees is relatively unstable. But on the whole, the performances of both are significantly better than UDS on all datastes, caused by the nature of supernode aggregation.

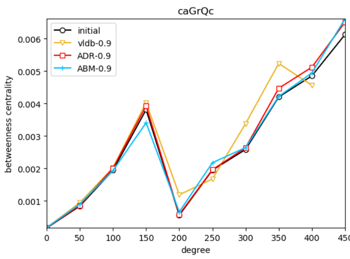


(a) p=0.9 (b) p=0.8 (c) p=0.7

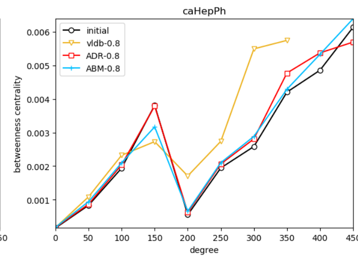
(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

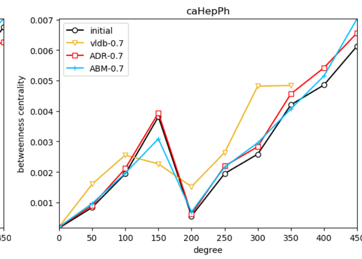
Fig. 7. Betweenness centrality of ca-GrQc



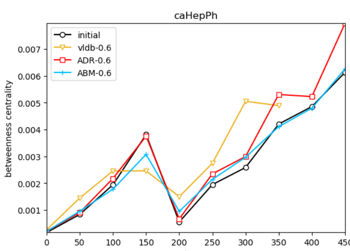
(a) p=0.9



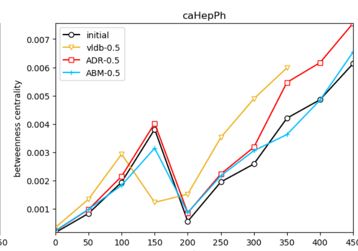
(b) p=0.8



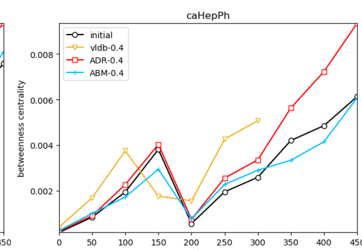
(c) p=0.7



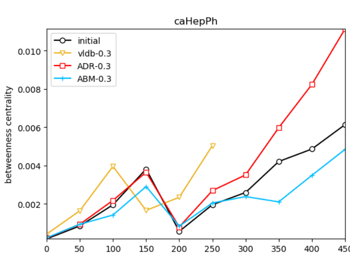
(d) p=0.6



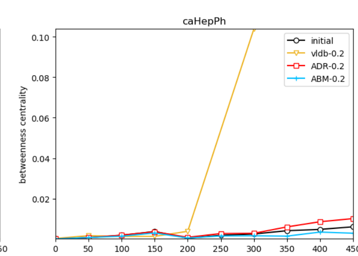
(e) p=0.5



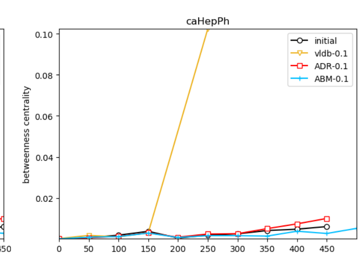
(f) p=0.4



(g) p=0.3

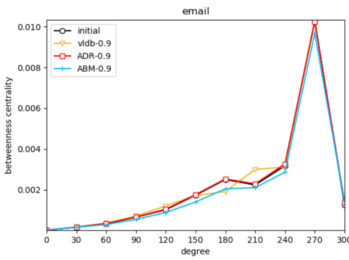


(h) p=0.2

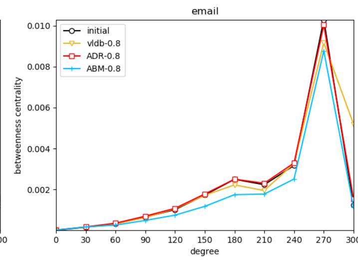


(i) p=0.1

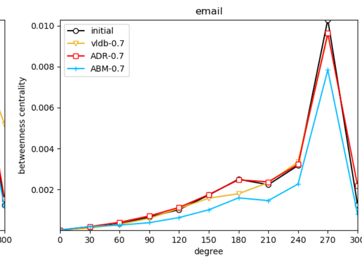
Fig. 8. Betweenness centrality of ca-HepPh



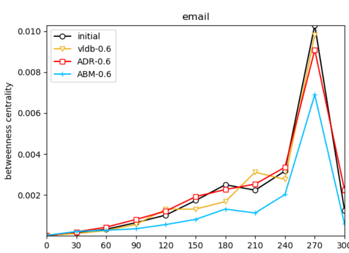
(a) p=0.9



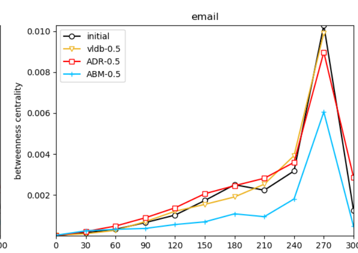
(b) p=0.8



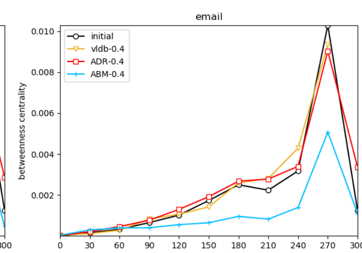
(c) p=0.7



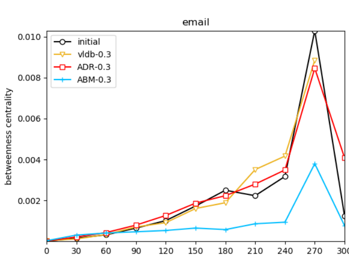
(d) p=0.6



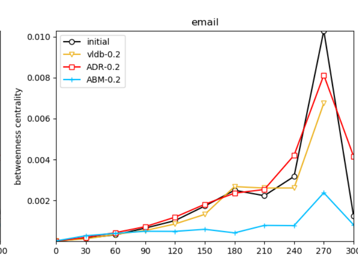
(e) p=0.5



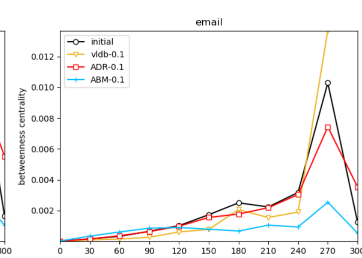
(f) p=0.4



(g) p=0.3



(h) p=0.2

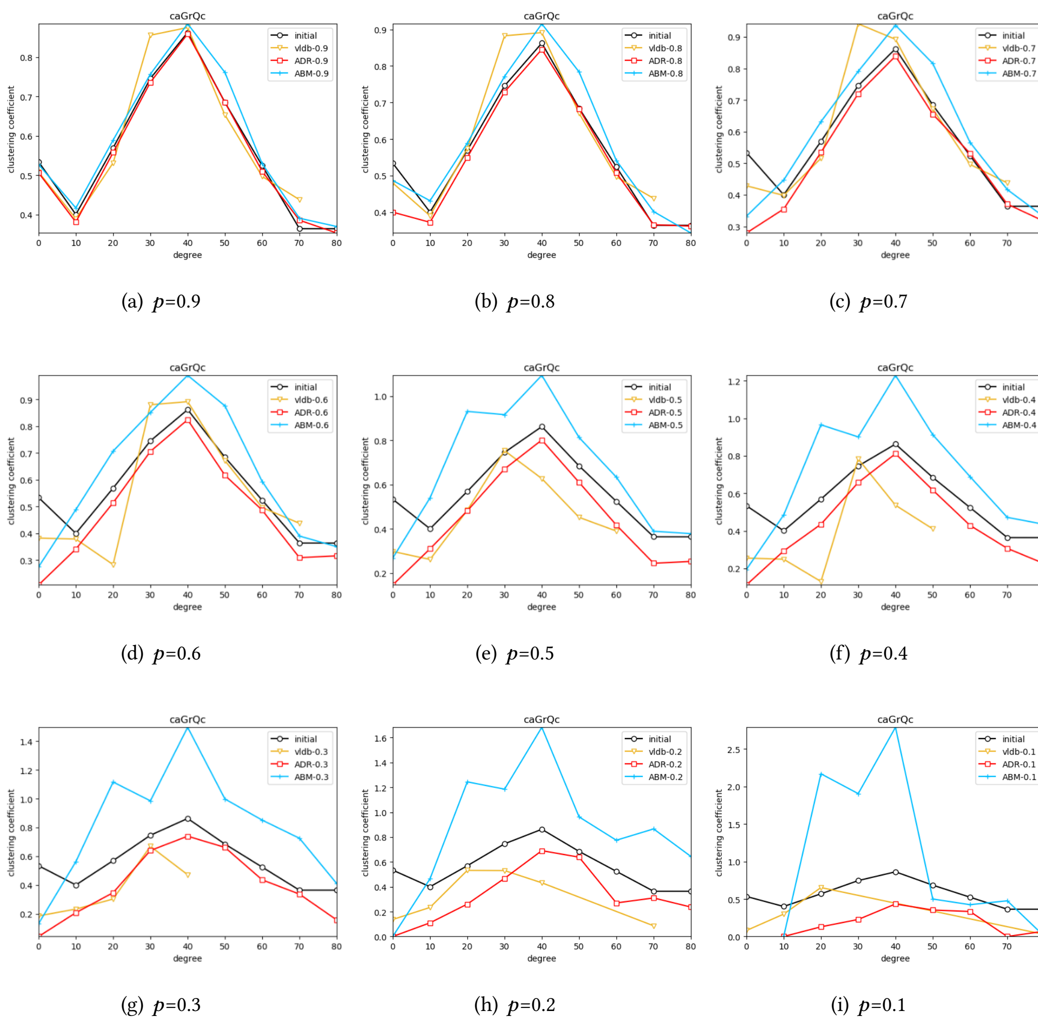


(i) p=0.1

Fig. 9. Betweenness centrality of email-Enron

2.4 Clustering coefficient

Figures 10-12 illustrate the clustering coefficient versus the vertex degree. The results are consistent with the results of shortest-path distance. For ca-GrQc, when p is larger, ADR and ABM are accurate in estimating the original graph from Figures 10(a)-10(d). When p is smaller, ADR performs best from Figures 10(e)-10(i). For all datasets, we can find that ADR performs best on ca-GrQc and email-Enron datasets while ABM performs best on ca-HepPh dataset. This shows that the two methods have different performances in characteristics extraction of different datasets.

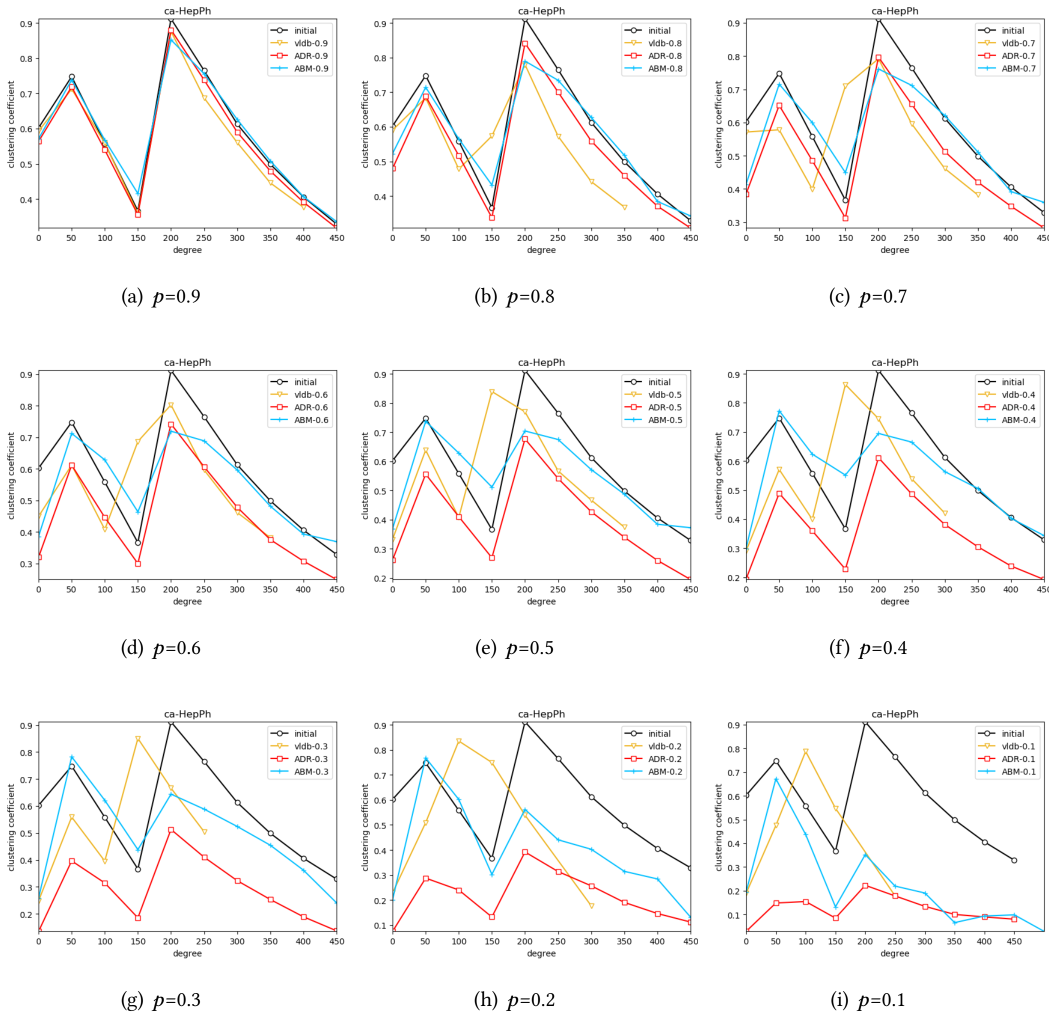


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 10. Clustering coefficient of ca-GrQc

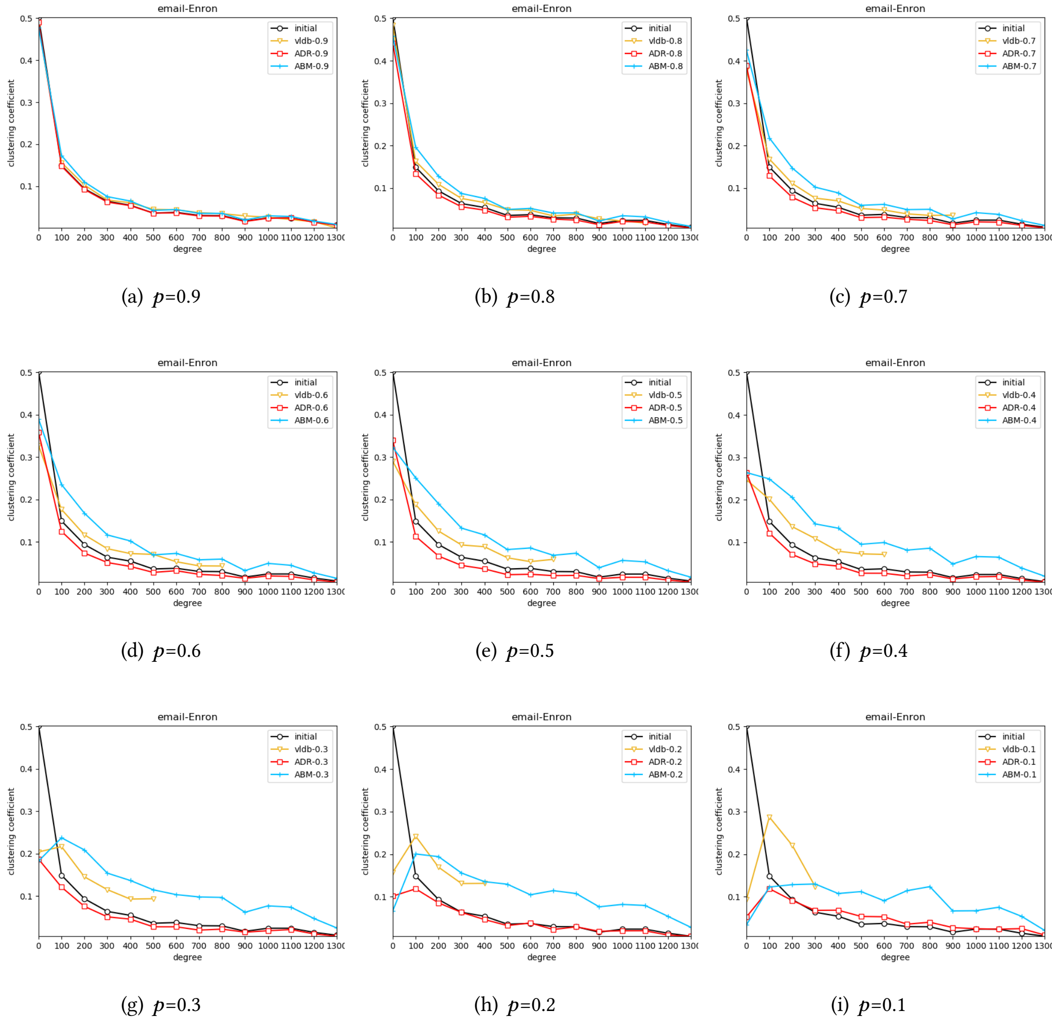


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 11. Clustering coefficient of ca-HepPh



(a) p=0.9 (b) p=0.8 (c) p=0.7

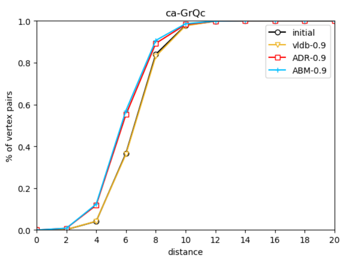
(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

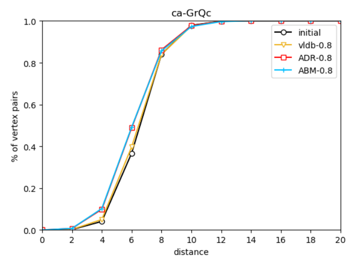
Fig. 12. Clustering coefficient of email-Enron

2.5 Hop-plot

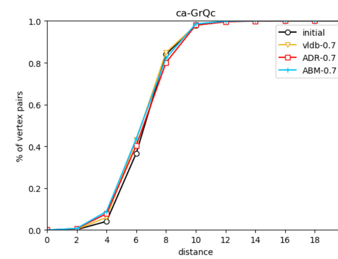
Figures 13-15 illustrate the hop-plot distribution. On all datasets, the estimations of the three methods for the initial graph are slightly different in different regions. We can see that ADR and ABM measure the hop-plot in larger distance region more accurately while UDS performs better in smaller distance region. But three of them perform great on the whole.



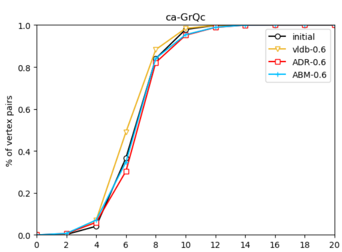
(a) p=0.9



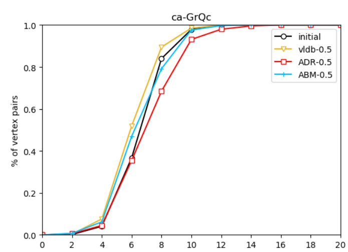
(b) p=0.8



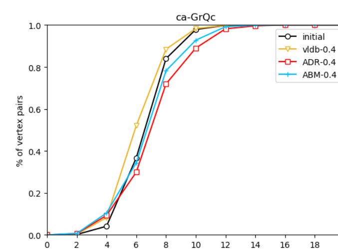
(c) p=0.7



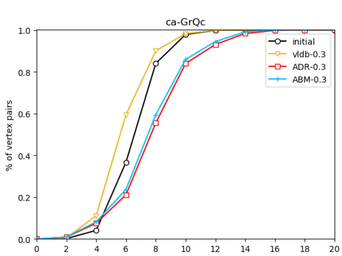
(d) p=0.6



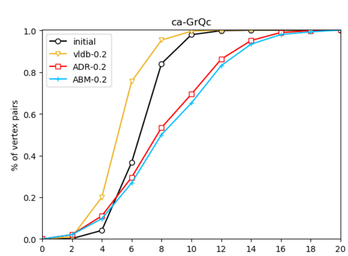
(e) p=0.5



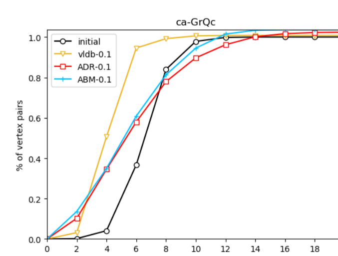
(f) p=0.4



(g) p=0.3

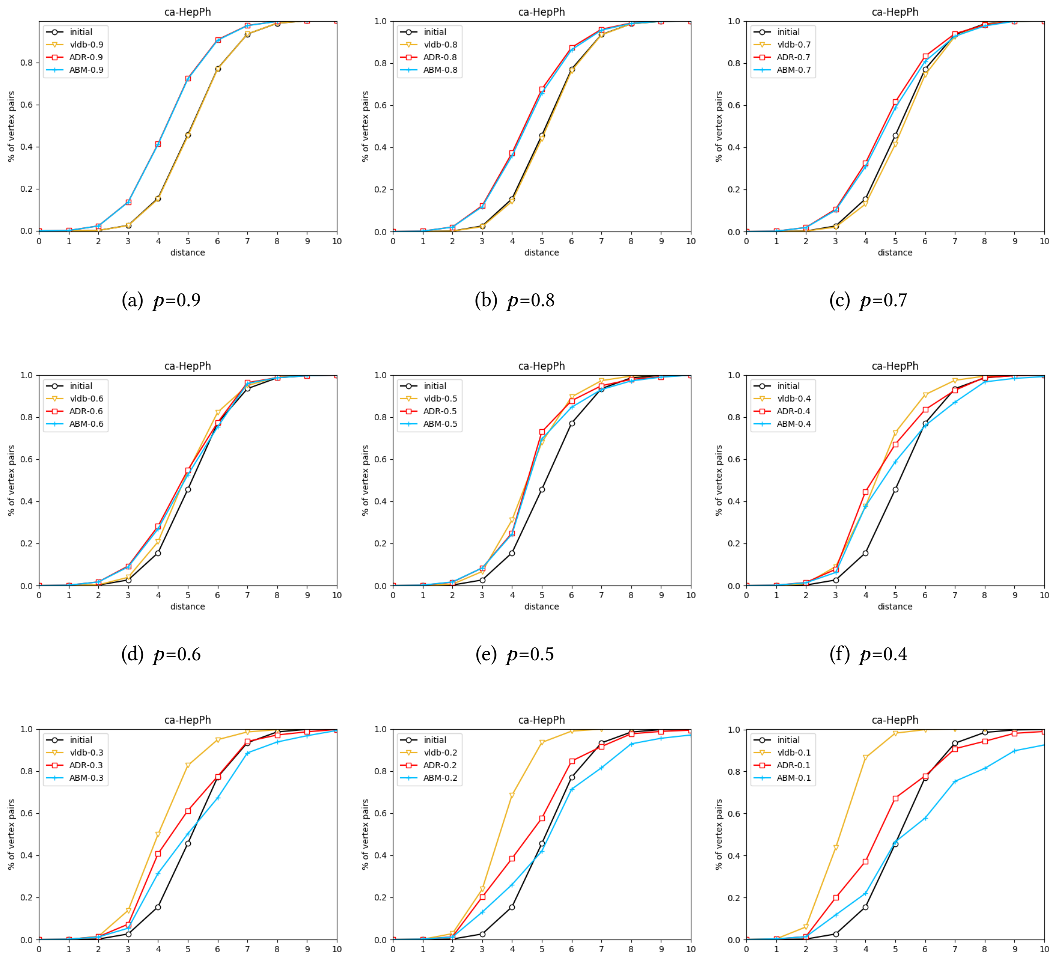


(h) p=0.2



(i) p=0.1

Fig. 13. Hop-plot of ca-GrQc

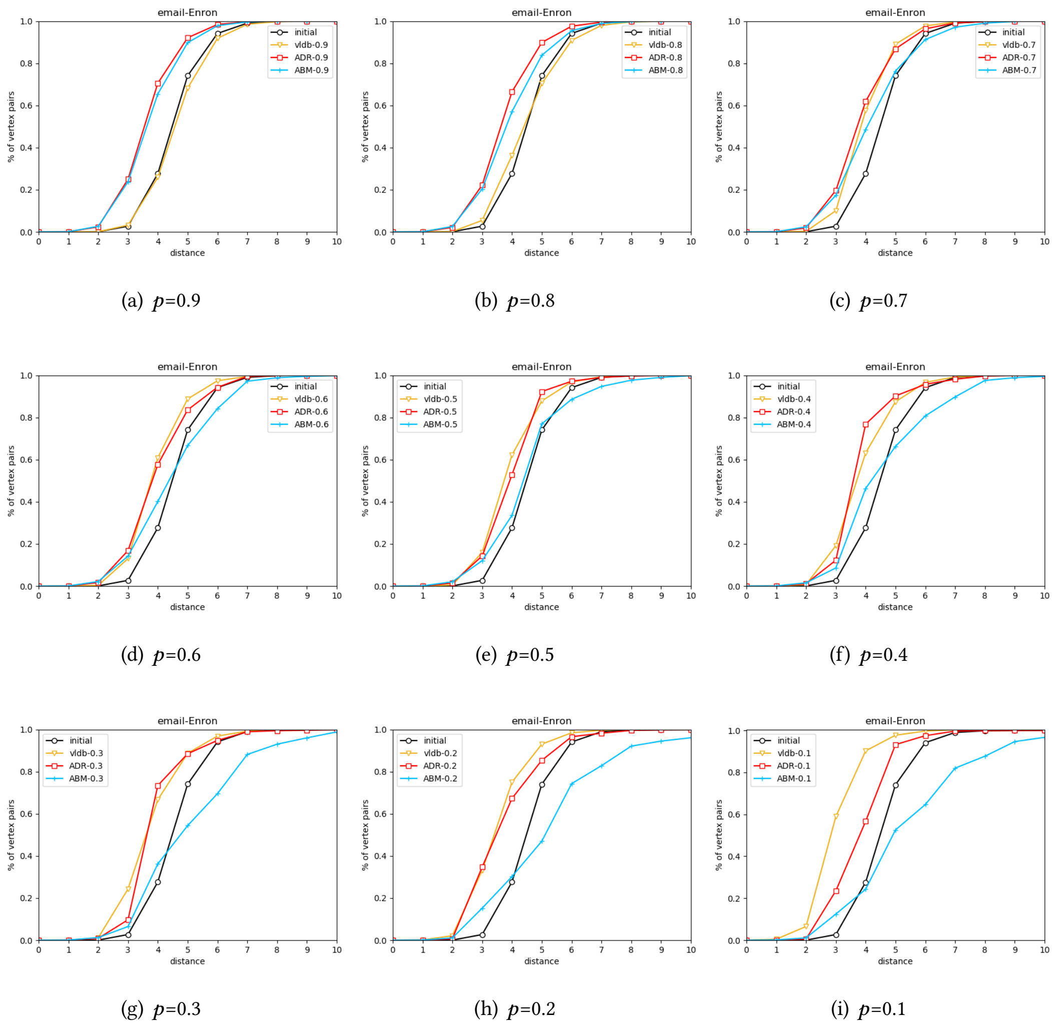


(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 14. Hop-plot of ca-HepPh



(a) p=0.9 (b) p=0.8 (c) p=0.7

(d) p=0.6 (e) p=0.5 (f) p=0.4

(g) p=0.3 (h) p=0.2 (i) p=0.1

Fig. 15. Hop-plot of email-Enron

2.6 Top-k Query

Tables 14-15 shows the utility results of our experiments where we compare ADR, ABM, and UDS with respect to top-k query, and t is set to 10. From the table we can see that the performance of ADR is excellent since it can reach almost 60% utility when p is reduced to 0.3 on all datasets. ABM's performance is also good, rank only second to ABM. When p is 0.1, the utility of UDS belows 0.2, which means it lost a lot of information. It is worth mentioning that, on large dataset such as com-LiveJournal, ADR and ABM both perform fantastic with utilities greater than 75% even though p is 0.1, demonstrating the proposed methods' power again for large dataset.

Table 14. Utility of Top-10% *I*

p	ca-GrQc			ca-HepPh		
	UDS	ADR	ABM	UDS	ADR	ABM
0.9	0.876	0.966	0.935	0.947	0.978	0.943
0.8	0.735	0.937	0.908	0.927	0.963	0.917
0.7	0.611	0.916	0.870	0.867	0.941	0.887
0.6	0.571	0.863	0.828	0.609	0.917	0.851
0.5	0.498	0.809	0.702	0.419	0.865	0.756
0.4	0.443	0.731	0.693	0.320	0.838	0.733
0.3	0.370	0.681	0.586	0.230	0.772	0.684
0.2	0.269	0.500	0.460	0.151	0.685	0.604
0.1	0.174	0.313	0.254	0.092	0.514	0.439

Table 15. Utility of Top-10% *II*

p	email-Enron			com-LiveJournal		
	UDS	ADR	ABM	UDS	ADR	ABM
0.9	0.775	0.966	0.885		0.963	0.984
0.8	0.537	0.939	0.798		0.900	0.986
0.7	0.357	0.898	0.750		0.856	0.976
0.6	0.283	0.859	0.696		0.823	0.957
0.5	0.226	0.812	0.595		0.797	0.938
0.4	0.180	0.761	0.572		0.776	0.913
0.3	0.141	0.698	0.543		0.725	0.870
0.2	0.105	0.586	0.454		0.642	0.850
0.1	0.075	0.394	0.292		0.787	0.893

2.7 Link prediction

Table 16 shows the utility results of link prediction. In order to reduce the influence of link prediction methods, we select Node2vec to generate models using graph embedding, and then use K-means to classify nodes on the models. The classification result is the basis of link prediction. Among them, we set the parameter p to 1, q to 1 of Node2vec, $n_c lusters$ to 5 of K-means. The results show that the performances of link prediction are different on various datasets. For ca-GrQc, all three techniques perform similarly. But for ca-HepPh and email-Enron, the utility of UDS drops too fast, which is obviously worse than ADR and ABM.

Table 16. Utility of Link Prediction

p	ca-GrQc			ca-HepPh			email-Enron		
	UDS	ADR	ABM	UDS	ADR	ABM	UDS	ADR	ABM
0.9	0.772	0.748	0.797	0.865	0.865	0.897	0.748	0.888	0.888
0.8	0.701	0.732	0.75	0.898	0.853	0.845	0.566	0.872	0.778
0.7	0.7	0.664	0.682	0.805	0.824	0.828	0.556	0.838	0.664
0.6	0.631	0.626	0.659	0.665	0.807	0.772	0.494	0.816	0.6
0.5	0.617	0.634	0.597	0.516	0.755	0.717	0.46	0.784	0.602
0.4	0.559	0.57	0.541	0.447	0.694	0.647	0.472	0.742	0.538
0.3	0.529	0.485	0.463	0.423	0.648	0.602	0.448	0.69	0.506
0.2	0.452	0.483	0.426	0.401	0.57	0.545	0.444	0.634	0.486
0.1	0.445	0.419	0.434	0.329	0.531	0.495	0.442	0.56	0.484