Disclaimer

This document is an exam summary that follows the script of the *Principles of Distributed Computing* lecture at ETH Zurich. The contribution to this is a list of algorithms with their most important properties. This summary was created during the spring semester 2020. Due to updates to the syllabus content, some material may no longer be relevant for future versions of the lecture.

I do not guarantee correctness or completeness, nor is this document endorsed by the lecturers. Feel free to point out any erratas. For the full LaTeX source code, consider `github.com/ymerkli/eth-summaries`.

chapter	Problem	local/congest	topology	algorithm & comments	time complexity	msg complexity	synchronous	needs n	needs IDs	needs leader	randomized
1	vertex coloring	L	all	Reduce, $(\Delta+1)$-coloring	$\mathcal{O}(n)$	$\mathcal{O}(2n\Delta)$	x		x		
1	vertex coloring	L	R tree	"Slow tree coloring", 2-coloring	$\mathcal{O}(D)=\mathcal{O}(n)$	$\mathcal{O}(n^2)$	x		x		
1	vertex coloring	L	R tree	"6-color", 6-coloring	$\mathcal{O}(\log^* n)$		x		x		
1	vertex coloring	L	R tree	"Six-2-three", 3-coloring	$\mathcal{O}(\log^* n)$	x	x	x			
1	vertex coloring	L	ring	Ex1.3a) non-uniform b) uniform	$\mathcal{O}(\log^* n)$		x		x		
2	Inform nodes, collect info	L	all	"Flooding/ Echo"	$\mathcal{O}(D)$	$\mathcal{O}(m+n)$					
2	BFS-construction	L	all	Synchronous performing of "flooding/echo" creates a BFS	$\mathcal{O}(n)$	$\mathcal{O}(m)$					
2	BFS-construction	L	all	Dijkstra BFS. Initiated by single node	$\mathcal{O}(D^2)$	$\mathcal{O}(m+nD)$					
2	BFS-construction	L	all	Bellman-Ford BFS. Initiated by single node	$\mathcal{O}(D)$	$\mathcal{O}(nm)$					
2	BFS-construction	L	all	Best "tradeoff" algorithm for BFS construction	$\mathcal{O}(D\log^3 n)$	$\mathcal{O}(m+n\log^3 n)$					
2	MST construction	L	all	"GHS". All nodes start synchronously. Creates ST minimizing edge weights. (no weights? Set all weights=1)	$\mathcal{O}(n\log n)$	$\mathcal{O}(m\log n)$			x		
2	Pair Matchup in Tree	L	R tree	Ex2.1a) Match up nodes in a tree as pairs with Echo	$\mathcal{O}(D)$	$\mathcal{O}(n)$	x				
3	vertex coloring	L	all	"Linial's coloring algorithm", $\mathcal{O}(\Delta^2\log\Delta)$-coloring	$\mathcal{O}(\log^* n)$		x		x		
3	vertex coloring	L	all	"Kuhn-Wattenhofer coloring algorithm", $(\Delta+1)$-coloring	$\mathcal{O}(\Delta\log\Delta+\log^* n)$		x		x		
3	vertex coloring	L	all	"Kuhn-Wattenhofer algorithm via defective coloring", $(\Delta+1)$-coloring	$\mathcal{O}(\Delta+\log^* n)$		x		x		
4	vertex coloring	L	UR tree	"Iterated peeling", 3-coloring	$\mathcal{O}(\log n)$		x				
5	MIS	L	all	"Slow MIS algorithm"	$\mathcal{O}(n)$	$\mathcal{O}(m)$	x		x		
5	MIS	L	all	"Luby's MIS algorithm", *non-deterministic*	whp $\mathcal{O}(\log n)$		x				x
5	vertex coloring <u>from</u> MIS	L	all	Coloring from MIS: finds a $(\Delta+1)$-coloring, *non-deterministic*	whp $\mathcal{O}(\log n)$		x				x
5	vertex coloring <u>to</u> MIS	L	all	Coloring to MIS: Given a C-coloring in time T, constructs MIS in time:	$\mathcal{O}(T+C)$						x
6	network decomp.	L	all	"Rozhon-Ghaffari algorithm", $(C,D)=(\mathcal{O}(\log n),\mathcal{O}(\log n))$ decomp.	*poly*$(\log n)$		x		x		
6	network decomp.	L	all	Randomized network decomp. algorithm, $(C,D)=(\mathcal{O}(\log n),\mathcal{O}(\log n))$, *non-deterministic*	whp $\mathcal{O}(\log n)$		x		x		x

chapter	Problem	local/congest	topology	algorithm & comments	time complexity	msg complexity	synchronous	needs n	needs IDs	needs leader	randomized
7	shared objects	L	all	"Arrow" algorithm, needs ST, complexities for "find" operation	$\mathcal{O}(D)$	$\mathcal{O}(D)$					
7	shared objects	L	all	"Read/write caching". Needs underlying ST.	-	3-competitive					
7	shared objects	L	all	"Ivy"-Algorithm. Needs underlying ST, complete graph. Complexities for the"find" operation:	avg. $\mathcal{O}(\log n)$						
8	sorting	C	array	"odd even sort" for Array	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$					
8	sorting	C	grid	"Shearsort" for mesh	$\mathcal{O}(\sqrt{n}(\log(n)+1))$		x	x			
8	sorting	C	sorting network	"Batcher"	$\mathcal{O}(\log^2 n)$		x				
9	diameter	C	all	Diameter via APSP	$\mathcal{O}(D)$		x			x	
10	MST construction	C	all, $(D{<<}n)$	Improved Boruvka MST algorithm	whp $\mathcal{O}((D+\sqrt{n})\log n)$		x				x
11	minimum cut	C	all	$(2+\epsilon)$ approximation algorithm for min. cut	$\mathcal{O}(K'(D+\sqrt{n})\log^2 n)$						
12	routing	C	all-to-all	Distributed routing problem, using random relay, $\mathcal{O}(\frac{n}{\log n})$ messages per node	$\mathcal{O}(1)$						x
13	Non-uniform init	L	wireless	repeatedly select leader, e.g. with Slotted Aloha	$\mathcal{O}(n)$		x	x			x
13	Uniform init w/ CD	L	wireless	Random repeated splitting into 2 groups	$\mathcal{O}(n)$		x				x
13	Uniform init wo/ CD	L	wireless	Needs a leader, overhead of factor 2	$\mathcal{O}(n)$		x				x
13	Non-uniform leader election w/ CD	L	wireless	"Slotted Aloha"	whp $\mathcal{O}(\log n)$		x	x			x
13	Uniform leader election wo/ CD	L	wireless	Exponential backoff algorithm (13.10)	whp $\mathcal{O}(\log^2 n)$		x				x
13	Uniform leader election w/ CD	L	wireless	Fast "Uniform leader election with CD"-Algorithm	whp $\mathcal{O}(\log n)$		x				x
13	Uniform leader election w/ CD	L	wireless	Even faster "fast uniform leader election". Elects leader wp $1-\frac{\log\log n}{\log n}$	wp $\mathcal{O}(\log\log n)$		x				x