Fermi ($b = 0.05$)		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.45	0.95	0.57	0.57	1.53	1.39	1.03	1.03	1.49	1.59	1.24	1.24
	16 (48)	5.16	2.33	1.17	1.17	5.44	3.35	2.16	2.14	5.35	4.20	3.10	3.06
	24 (108)	10.93	4.04	1.61	1.76	11.51	5.63	3.19	3.15	11.54	6.95	4.65	4.55
	32 (192)	20.17	6.80	2.45	2.43	21.06	8.82	4.66	4.56	19.97	10.27	6.25	6.09
$m = 1725$	40 (300)	29.37	8.90	2.79	3.41	30.84	11.98	5.73	5.66	31.89	14.24	8.01	7.92
75×75	16 (48)	16.18	7.02	3.69	3.49	17.17	10.54	7.14	6.63	17.77	13.42	10.14	9.40
	24 (108)	36.46	13.16	6.21	5.65	36.38	18.01	11.33	10.04	37.32	23.22	16.79	14.66
	32 (192)	64.89	20.30	8.25	7.23	63.27	26.67	15.71	13.35	68.13	37.46	27.68	21.03
	40 (300)	109.39	37.03	15.48	9.78	107.53	47.74	30.95	19.70	99.48	47.97	36.35	25.35
$m = 4\,396$	48 (432)	152.25	48.46	22.61	13.63	153.12	60.20	41.17	24.96	150.91	69.15	53.81	34.69
100×100	48 (432)	58.25	24.76	5.82	4.74	57.52	28.84	11.00	8.87	63.85	35.13	17.67	14.37
	56 (588)	80.97	34.98	7.66	5.64	80.68	39.95	15.16	11.51	81.57	46.33	23.95	17.48
	64 (768)	112.28	51.18	11.69	7.34	115.51	61.67	27.69	18.99	121.67	70.48	39.93	26.91
	72 (972)	156.94	70.36	16.24	9.15	152.45	77.67	32.77	22.37	146.05	80.07	44.72	30.42
$m = 6\,630$	80 (1200)	177.53	79.25	18.55	11.06	178.88	88.61	37.62	24.92	184.10	95.71	51.76	35.85
125×125	64 (768)	566.74	161.87	57.54	41.41	558.05	193.76	113.56	81.03	576.05	233.79	192.25	121.89
	72 (972)	694.20	194.31	74.20	46.27	711.64	230.42	142.64	89.32	690.31	265.72	217.76	133.85
	80 (1200)	894.75	238.60	93.88	60.67	865.25	276.34	174.91	107.87	858.70	315.72	258.99	157.75
	88 (1452)	1021.17	266.79	96.21	61.80	1016.82	307.52	182.69	112.86	1026.53	349.38	266.17	163.10
$m = 10\,293$	96 (1728)	1242.27	310.09	99.37	64.57	1230.02	350.96	195.12	121.05	1215.55	393.79	303.39	174.94
150×150	80 (1200)	844.55	284.84	82.25	62.76	920.81	342.88	166.42	121.57	898.16	373.48	234.40	167.74
	88 (1452)	1054.82	341.18	90.67	68.76	1051.97	382.23	179.08	132.47	1060.37	423.40	255.13	183.45
	96 (1728)	1223.53	368.43	93.59	71.75	1151.56	400.01	168.31	127.55	1208.25	474.35	268.44	193.48
	104 (2028)	1508.65	429.07	106.06	82.15	1372.15	459.62	178.90	132.34	1431.71	531.42	303.22	215.92
$m = 19\,244$	112 (2352)	1603.52	476.35	108.64	83.81	1582.70	526.15	213.65	157.43	1735.34	606.89	330.01	232.02
175×175	120 (2700)	5296.95	1263.40	430.95	279.63	5203.39	1451.38	883.90	511.66	5186.49	1587.85	1229.33	684.29
	128 (3072)	5888.65	1417.79	481.36	297.38	5904.74	1596.53	909.78	532.06	5848.29	1783.08	1359.77	755.42
	136 (3468)	6679.18	1567.28	501.13	311.95	6678.50	1789.09	1015.70	580.48	6722.86	1968.25	1458.17	805.58
	144 (3888)	7565.02	1757.60	585.67	363.70	7489.12	1903.67	995.32	580.57	7435.71	2107.83	1516.09	827.14
$m = 23\,954$	152 (4332)	8269.23	1862.62	565.10	350.25	8054.61	2049.62	1081.18	620.53	8184.49	2231.23	1537.52	845.30
200×200	208 (8112)	4888.16	1847.63	246.86	187.38	4946.87	1948.57	543.72	384.57	4959.96	1970.62	648.57	456.69
	216 (8748)	5311.76	1830.34	224.95	176.30	5319.10	1941.68	409.20	310.41	5109.65	1973.49	576.04	418.92
	224 (9408)	5478.13	1947.72	210.21	164.79	5042.59	1933.12	407.88	310.45	5436.63	2098.99	610.84	441.06
	232 (10092)	6026.48	2099.21	227.80	178.86	5773.85	2107.06	414.53	313.21	6345.89	2268.95	630.99	463.34
$m = 32\,612$	240 (10800)	5890.26	2090.45	219.58	178.58	5812.73	2157.52	410.60	309.98	5961.60	2312.58	627.36	459.43

Table 1: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using **Fermi** function (transition curve) with $\mathbf{b = 0.05}$.

Fermi ($b = 0.1$)		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.42	0.87	0.50	0.50	1.49	1.34	0.96	0.96	1.62	1.69	1.31	1.31
	16 (48)	5.52	2.34	1.11	1.11	5.47	3.27	2.07	2.04	5.48	4.20	3.03	3.00
	24 (108)	11.36	4.29	1.75	1.74	11.47	5.81	3.39	3.34	11.58	7.14	4.75	4.62
	32 (192)	20.29	6.76	2.33	2.33	19.33	8.32	4.20	4.13	20.78	10.82	6.66	6.49
$m = 1725$	40 (300)	30.82	9.31	2.79	2.85	31.37	12.19	5.76	5.55	31.01	14.51	8.32	7.99
75×75	16 (48)	17.09	7.16	3.67	3.47	17.49	10.91	7.45	6.90	17.86	13.56	10.26	9.28
	24 (108)	37.97	14.12	6.26	5.49	37.73	19.46	12.15	10.34	39.16	24.29	17.85	14.85
	32 (192)	69.03	24.11	9.63	7.18	68.66	31.46	19.74	14.49	68.71	37.55	28.16	21.18
	40 (300)	109.50	37.75	17.17	15.51	109.14	46.95	31.48	19.96	109.94	55.62	46.18	30.22
$m = 4\,396$	48 (432)	156.01	49.80	20.37	12.03	151.31	57.87	37.31	23.18	153.78	71.45	58.95	37.85
100×100	48 (432)	75.28	36.33	9.90	7.81	64.99	37.48	16.18	10.87	68.39	43.49	24.14	16.92
	56 (588)	92.54	44.60	11.42	6.86	93.82	50.09	21.74	14.82	88.34	53.62	29.36	20.94
	64 (768)	113.36	54.04	12.93	8.14	114.87	60.15	25.04	16.76	107.60	61.10	31.32	21.87
	72 (972)	145.46	68.28	17.58	10.97	138.65	69.16	27.38	17.78	131.59	72.64	36.85	25.07
$m = 6\,630$	80 (1200)	163.93	73.64	17.05	10.54	172.86	82.95	33.95	22.88	176.77	93.24	49.79	34.55
125×125	64 (768)	500.62	150.68	61.76	38.53	528.04	182.91	117.61	74.52	519.99	210.42	172.47	107.70
	72 (972)	646.99	182.29	69.28	43.09	659.93	217.11	133.66	84.42	654.67	251.95	200.27	123.89
	80 (1200)	833.13	226.18	87.17	54.15	822.08	263.89	162.49	101.42	793.28	288.91	221.93	136.57
	88 (1452)	992.09	261.02	92.79	57.86	989.98	301.85	171.10	106.02	980.01	336.52	248.69	151.21
$m = 10\,293$	96 (1728)	1153.60	293.71	102.97	64.88	1226.15	350.66	200.04	123.95	1180.10	385.91	274.61	166.80
150×150	80 (1200)	867.17	278.15	75.10	56.56	776.67	298.99	135.19	99.50	825.49	346.22	207.25	150.24
	88 (1452)	985.66	319.84	86.05	65.10	899.56	352.31	163.59	120.23	1016.81	414.43	252.49	178.70
	96 (1728)	1209.18	370.29	96.83	72.60	1194.25	413.76	174.79	129.01	1263.41	473.79	266.11	190.69
	104 (2028)	1475.17	432.88	101.93	78.22	1338.37	466.25	203.18	150.82	1419.74	528.72	295.00	212.64
$m = 19\,244$	112 (2352)	1679.41	492.13	103.45	79.05	1667.48	548.61	227.39	166.30	1578.65	573.96	299.82	217.23
175×175	120 (2700)	5169.06	1250.52	434.40	270.26	5185.72	1447.29	896.52	515.22	5144.86	1592.83	1262.47	701.24
	128 (3072)	5768.63	1287.43	358.56	247.37	5825.08	1442.99	656.47	435.02	5791.02	1605.34	981.35	632.11
	136 (3468)	6493.98	1440.29	395.12	273.33	6370.77	1556.42	684.36	456.02	6499.32	1765.66	1056.06	670.71
	144 (3888)	7288.50	1553.24	387.32	271.95	7304.44	1796.92	862.95	543.85	7389.09	2077.66	1514.43	827.54
$m = 23\,954$	152 (4332)	8194.98	1840.73	566.81	352.13	8208.31	2035.08	1049.18	600.70	8222.40	2284.11	1623.13	895.14
200×200	208 (8112)	4388.35	1622.19	187.58	145.83	4767.74	1773.62	362.51	272.41	4383.47	1769.99	539.21	394.15
	216 (8748)	5301.45	1963.83	275.58	210.94	5166.28	2076.74	573.92	401.98	5486.74	2207.37	883.91	587.70
	224 (9408)	5530.34	2005.56	261.39	197.84	5136.07	2020.83	496.25	351.13	5994.79	2357.12	994.41	645.69
	232 (10092)	6364.13	2301.04	341.47	252.49	5721.22	2250.16	586.80	413.79	6240.72	2414.75	889.92	593.14
$m = 32\,612$	240 (10800)	6923.98	2454.04	332.10	244.36	6449.02	2478.61	637.10	451.19	6068.20	2517.47	877.29	589.28

Table 2: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using **Fermi** function (transition curve) with $\mathbf{b = 0.1}$.

Fermi ($b = 0.2$)		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.48	0.92	0.54	0.53	1.55	1.34	0.96	0.96	1.57	1.64	1.28	1.27
	16 (48)	5.31	2.42	1.19	1.19	5.34	3.31	2.14	2.12	5.40	4.13	3.00	2.95
	24 (108)	11.15	4.16	1.64	1.64	11.70	5.64	3.13	3.10	11.79	7.41	5.02	4.95
	32 (192)	20.02	6.57	2.25	2.24	20.59	8.94	4.64	4.51	20.17	10.80	6.67	6.47
$m = 1725$	40 (300)	29.22	9.07	2.71	2.72	33.10	12.94	6.09	5.70	32.21	14.73	8.25	7.97
75×75	16 (48)	16.42	6.92	3.67	3.35	16.59	10.17	7.29	6.51	17.06	13.18	10.49	9.13
	24 (108)	36.61	14.09	6.64	5.63	37.48	19.62	13.07	10.80	37.45	23.18	17.48	14.49
	32 (192)	66.40	22.29	9.19	7.28	64.96	29.18	18.27	13.77	66.96	35.82	26.23	19.72
	40 (300)	107.35	35.63	14.61	9.33	100.99	40.86	24.54	17.73	104.55	50.77	37.76	26.53
$m = 4\,396$	48 (432)	150.13	46.74	18.85	11.48	150.65	57.47	35.61	23.56	148.10	65.14	47.71	33.21
100×100	48 (432)	61.64	32.25	8.97	5.79	72.58	39.70	17.68	12.10	61.54	38.58	20.59	17.48
	56 (588)	93.09	40.09	9.82	6.47	84.22	44.73	18.49	13.14	83.52	50.42	28.03	20.40
	64 (768)	115.96	51.31	11.83	7.53	117.81	58.37	24.34	17.09	113.64	63.94	33.26	23.95
	72 (972)	152.47	67.65	16.03	10.00	145.36	73.67	31.34	21.97	139.98	75.21	37.31	26.59
$m = 6\,630$	80 (1200)	170.80	73.42	16.64	10.67	177.34	83.70	32.43	23.13	176.42	91.99	45.92	32.62
125×125	64 (768)	525.13	153.31	58.46	36.03	489.37	176.14	112.49	71.95	528.50	213.79	176.88	110.59
	72 (972)	666.23	188.03	74.78	46.56	666.92	224.93	143.89	89.93	676.63	253.43	201.67	124.07
	80 (1200)	795.71	217.43	79.02	53.39	809.09	262.54	165.34	102.28	816.82	298.77	238.71	145.22
	88 (1452)	935.17	247.27	90.99	56.84	982.84	290.70	160.42	100.55	1014.06	342.78	254.00	155.78
$m = 10\,293$	96 (1728)	1173.15	299.95	101.43	64.45	1166.33	345.14	194.46	118.40	1190.03	386.00	279.44	167.61
150×150	80 (1200)	792.81	270.34	71.14	53.91	885.97	323.02	154.06	113.81	781.62	341.66	205.92	148.61
	88 (1452)	1087.09	347.07	95.70	72.76	1000.70	362.20	159.41	116.66	1004.40	411.16	237.13	171.97
	96 (1728)	1199.47	369.50	89.80	68.33	1150.31	402.25	171.20	124.87	1224.97	472.10	274.08	198.66
	104 (2028)	1322.68	414.94	97.84	75.21	1325.23	460.11	186.55	137.97	1415.41	523.32	285.40	204.74
$m = 19\,244$	112 (2352)	1630.76	491.56	105.73	81.13	1556.78	527.20	211.62	156.83	1461.54	554.25	293.84	207.13
175×175	120 (2700)	5155.96	1261.36	457.23	282.37	5145.92	1424.10	860.50	503.09	5169.54	1585.71	1258.24	695.11
	128 (3072)	5837.08	1405.66	491.64	306.28	5771.36	1575.91	918.85	531.87	5841.95	1755.81	1355.98	741.86
	136 (3468)	6543.28	1542.80	494.93	309.08	6552.35	1724.12	954.23	556.54	6565.03	1939.37	1471.05	818.55
	144 (3888)	7334.02	1679.64	511.28	322.47	7377.21	1894.25	1027.63	603.02	7330.43	2078.31	1516.19	830.34
$m = 23\,954$	152 (4332)	8203.19	1836.97	547.35	342.00	8257.45	2047.77	1052.04	621.60	8176.09	2282.52	1635.17	900.99
200×200	208 (8112)	4945.67	1883.07	285.60	212.02	4649.12	1861.02	529.19	377.75	5369.49	2131.25	861.71	569.01
	216 (8748)	5282.80	1964.88	292.08	220.28	5145.71	2042.81	568.19	398.19	5260.75	2086.96	771.16	503.09
	224 (9408)	5542.10	2098.60	296.15	225.09	5680.08	2206.60	608.63	425.94	4981.30	2177.07	837.41	563.62
	232 (10092)	6026.23	2208.52	300.68	230.31	5920.56	2310.07	604.61	427.09	6085.94	2418.61	867.80	578.08
$m = 32\,612$	240 (10800)	6587.79	2375.22	328.72	244.90	6153.20	2406.46	612.35	438.98	6273.71	2516.40	903.65	603.83

Table 3: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using **Fermi** function (transition curve) with $\mathbf{b = 0.2}$.

Fermi ($b = 0.4$)		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.47	0.96	0.56	0.56	1.59	1.41	1.01	1.02	1.62	1.75	1.37	1.37
	16 (48)	5.61	2.41	1.15	1.19	5.37	3.32	2.11	2.09	5.67	4.28	3.05	3.02
	24 (108)	12.49	4.41	1.72	1.72	12.07	6.00	3.38	3.31	12.79	8.46	5.94	5.29
	32 (192)	20.61	6.83	2.34	2.36	21.43	9.25	4.75	4.67	20.35	10.82	6.50	6.34
$m = 1725$	40 (300)	32.22	9.88	2.99	2.94	32.85	12.80	6.07	5.91	33.74	16.37	10.16	8.87
75×75	16 (48)	16.86	7.08	3.77	3.47	17.01	10.61	7.70	6.97	17.23	12.74	9.93	8.89
	24 (108)	37.81	13.89	6.59	5.73	37.54	18.40	11.74	9.90	37.60	23.85	18.21	15.06
	32 (192)	67.59	23.16	9.91	7.62	67.96	29.45	17.96	13.32	67.78	37.77	29.74	21.58
	40 (300)	103.88	32.57	13.33	9.28	93.54	35.96	20.44	17.10	96.28	43.98	30.38	25.67
$m = 4\,396$	48 (432)	135.52	36.73	12.72	10.38	139.54	47.47	25.20	20.45	139.13	57.73	38.17	31.00
100×100	48 (432)	66.94	31.49	8.16	5.38	70.20	38.05	16.38	11.57	63.96	38.88	21.58	15.61
	56 (588)	93.19	43.28	11.57	7.66	88.66	48.30	20.19	18.26	86.53	52.30	28.07	20.13
	64 (768)	114.92	52.83	11.82	7.13	119.61	62.76	25.42	16.88	112.06	63.96	33.08	23.63
	72 (972)	140.01	66.52	15.28	8.99	149.00	73.77	28.95	19.54	130.03	69.16	31.52	23.04
$m = 6\,630$	80 (1200)	157.11	66.84	13.24	8.61	166.19	75.67	25.29	18.12	160.76	82.12	39.77	30.47
125×125	64 (768)	549.39	156.05	59.70	36.31	507.70	182.59	119.69	74.84	546.80	216.76	176.82	108.55
	72 (972)	665.49	188.01	70.13	44.08	668.41	220.79	136.31	85.27	679.95	259.06	207.85	128.10
	80 (1200)	869.26	230.46	83.88	51.82	815.59	258.38	151.98	95.14	828.71	294.84	222.25	135.14
	88 (1452)	994.44	259.04	91.55	57.77	988.81	297.22	168.00	104.27	972.64	332.06	242.22	147.90
$m = 10\,293$	96 (1728)	1204.78	296.89	93.06	58.13	1179.80	340.19	189.92	117.32	1167.15	384.87	280.16	168.77
150×150	80 (1200)	743.55	257.08	68.93	51.86	915.39	341.05	170.34	124.65	873.41	361.05	224.07	159.22
	88 (1452)	949.05	312.98	81.86	63.05	1022.85	374.03	173.17	127.37	1027.14	418.91	255.05	180.68
	96 (1728)	1146.44	368.43	94.26	71.42	1147.80	417.61	188.90	140.80	1177.03	456.59	259.09	185.40
	104 (2028)	1325.19	414.85	99.79	76.98	1384.28	477.80	201.10	149.64	1355.33	515.99	294.04	209.03
$m = 19\,244$	112 (2352)	1510.11	470.76	109.52	83.03	1656.59	543.81	211.19	154.62	1747.33	615.61	334.41	239.22
175×175	120 (2700)	5118.49	1269.46	469.50	290.97	5075.76	1418.96	857.15	495.16	5109.13	1595.70	1252.90	693.06
	128 (3072)	5787.41	1416.35	495.15	307.72	5842.90	1575.94	909.58	526.87	5844.55	1761.23	1326.59	738.97
	136 (3468)	6443.71	1544.51	517.53	320.65	6516.24	1753.90	990.59	580.09	6564.53	1926.23	1435.69	781.14
	144 (3888)	7330.71	1700.50	534.85	334.37	7335.22	1907.00	1060.18	607.76	7310.63	2090.16	1516.11	835.51
$m = 23\,954$	152 (4332)	8199.14	1851.13	556.96	347.06	8173.53	2040.07	1027.23	593.40	8164.99	2256.49	1572.97	860.80
200×200	208 (8112)	4957.54	1851.43	246.41	182.16	4386.72	1829.69	515.30	374.12	4984.06	2060.17	801.07	538.65
	216 (8748)	5271.93	1968.75	288.01	218.05	4977.10	2038.22	615.14	422.13	5072.65	2145.84	833.84	551.20
	224 (9408)	5488.80	2059.76	300.12	229.57	5567.24	2152.67	590.04	414.37	5358.05	2243.21	855.39	564.76
	232 (10092)	6500.64	2292.12	338.13	253.38	5752.88	2276.21	603.21	425.28	6030.54	2447.69	942.28	623.06
$m = 32\,612$	240 (10800)	6787.51	2349.02	344.71	265.61	5960.08	2392.41	614.67	435.00	6632.60	2610.94	927.82	619.76

Table 4: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using **Fermi** function (transition curve) with $\mathbf{b = 0.4}$.

Fermi ($b = 0.8$)		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.60	1.01	0.59	0.58	1.55	1.32	0.92	0.92	1.68	1.77	1.36	1.36
	16 (48)	5.67	2.50	1.24	1.24	5.76	3.51	2.24	2.28	5.66	4.30	3.07	3.04
	24 (108)	12.30	4.53	1.85	1.85	12.39	6.04	3.38	3.36	12.12	7.30	4.74	4.67
	32 (192)	21.11	7.01	2.40	2.40	21.38	9.13	4.66	4.59	20.74	10.99	6.66	6.49
$m = 1725$	40 (300)	31.04	9.46	2.84	2.80	32.70	12.66	5.91	5.74	32.82	15.34	8.68	8.28
75×75	16 (48)	16.14	7.08	4.00	3.70	16.56	10.16	7.39	6.71	16.39	12.74	10.15	9.18
	24 (108)	34.71	12.19	5.76	5.25	35.40	17.12	11.33	10.11	35.26	21.32	16.04	14.19
	32 (192)	63.22	19.41	8.15	7.18	63.34	26.59	15.71	13.09	61.86	31.73	22.37	19.26
	40 (300)	95.01	27.28	10.37	8.89	95.85	35.55	19.91	16.91	97.37	43.47	29.24	24.41
$m = 4\,396$	48 (432)	139.93	38.10	13.21	10.54	144.50	51.38	28.46	21.85	137.50	55.81	35.78	29.07
100×100	48 (432)	60.33	26.68	6.28	4.97	57.31	31.15	12.39	9.71	62.13	36.68	18.46	14.68
	56 (588)	81.38	35.37	8.01	5.97	84.00	41.24	15.37	11.66	81.69	45.37	22.03	17.11
	64 (768)	105.26	45.91	9.23	6.38	100.86	49.16	16.81	12.18	104.90	57.09	26.54	20.48
	72 (972)	136.81	58.63	12.02	8.08	132.71	63.94	22.94	16.50	137.13	70.01	31.87	23.97
$m = 6\,630$	80 (1200)	162.84	66.67	13.34	9.11	162.31	74.45	24.92	18.01	164.69	83.21	38.00	28.19
125×125	64 (768)	547.06	158.20	64.76	40.73	543.91	187.74	118.01	74.50	552.46	219.50	183.55	112.87
	72 (972)	665.35	185.57	69.23	43.07	676.45	224.15	139.18	87.20	637.22	244.39	192.35	118.53
	80 (1200)	806.99	217.22	75.32	47.12	812.96	262.47	163.48	101.22	811.38	293.93	226.14	138.39
	88 (1452)	1001.14	258.83	84.53	53.99	996.92	305.92	183.65	115.01	992.31	336.56	246.56	151.35
$m = 10\,293$	96 (1728)	1162.88	299.41	104.23	66.37	1179.84	345.36	192.05	117.79	1228.15	399.67	289.17	176.34
150×150	80 (1200)	962.48	302.91	85.26	64.91	782.73	306.14	137.67	101.87	842.79	362.50	222.94	161.90
	88 (1452)	1055.76	326.79	86.80	66.51	1041.30	364.99	159.27	117.77	1011.36	400.80	237.08	170.41
	96 (1728)	1157.47	362.21	89.97	69.09	1155.02	413.16	184.53	133.40	1128.36	437.63	241.19	175.28
	104 (2028)	1436.51	417.62	99.52	76.69	1361.70	462.36	186.49	138.46	1339.84	504.27	278.42	201.77
$m = 19\,244$	112 (2352)	1600.14	485.92	108.88	84.74	1535.02	517.88	206.66	154.21	1528.41	573.52	300.07	216.52
175×175	120 (2700)	5065.24	1258.68	462.63	286.86	5120.81	1441.29	886.22	509.17	5108.55	1576.66	1204.18	672.50
	128 (3072)	5794.74	1416.47	490.05	305.61	5791.75	1588.52	925.66	539.39	5820.61	1771.46	1337.58	736.75
	136 (3468)	6516.18	1564.39	530.81	328.76	6566.59	1755.26	997.33	571.52	6513.39	1896.60	1345.36	745.49
	144 (3888)	7270.21	1684.36	520.72	325.06	7326.62	1909.39	1049.11	602.78	7298.72	2081.84	1490.78	819.72
$m = 23\,954$	152 (4332)	8126.46	1856.12	565.49	352.85	8153.81	2075.20	1104.91	627.89	8138.71	2280.20	1631.39	892.12
200×200	208 (8112)	4936.79	1862.33	284.47	214.72	4707.18	1878.63	519.21	357.99	5072.37	2094.38	843.92	560.39
	216 (8748)	5199.94	1936.78	275.79	213.45	5342.87	2034.71	560.83	396.60	5164.89	2148.63	819.27	555.59
	224 (9408)	5732.88	2135.40	299.67	230.91	5059.03	2158.42	575.78	406.33	5570.17	2312.11	897.78	595.48
	232 (10092)	6094.73	2243.29	303.94	227.64	5609.15	2298.16	593.96	436.35	6289.70	2466.40	927.13	627.30
$m = 32\,612$	240 (10800)	6030.58	2260.69	299.01	219.01	6809.19	2509.47	649.57	469.12	7023.87	2688.74	1032.77	695.42

Table 5: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using **Fermi** function (transition curve) with $\mathbf{b = 0.8}$.

Exponential		Nb. of buoys displaced: 1				Nb. of buoys displaced: 2				Nb. of buoys displaced: 3			
			Update procedures				Update procedures				Update procedures		
Grid size	N ($\lvert\Xi_\omega\rvert$)	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3	F-scratch	V1	V2	V3
50×50	8 (12)	1.46	0.91	0.53	0.52	1.55	1.35	0.99	0.98	1.63	1.71	1.35	1.33
	16 (48)	5.21	2.34	1.12	1.11	5.19	3.12	1.97	1.94	5.45	4.11	2.99	2.92
	24 (108)	11.59	4.29	1.74	1.72	11.77	5.92	3.38	3.30	11.80	7.27	4.86	4.70
	32 (192)	19.73	6.43	2.14	2.11	20.71	8.94	4.59	4.44	20.72	10.69	6.47	6.24
$m = 1725$	40 (300)	32.10	9.76	2.96	2.85	31.69	12.09	5.55	5.34	31.73	14.72	8.44	8.02
75×75	16 (48)	14.78	6.55	3.73	3.42	15.41	9.29	6.69	6.04	16.22	11.81	9.51	8.52
	24 (108)	34.59	12.08	5.79	5.20	34.74	16.53	10.78	9.44	34.63	20.84	16.06	13.99
	32 (192)	60.63	18.46	7.46	6.55	59.74	24.82	14.68	12.26	67.93	37.10	28.34	20.64
	40 (300)	103.36	34.92	14.50	9.05	101.29	41.44	26.07	18.09	101.47	51.47	40.54	27.70
$m = 4\,396$	48 (432)	148.80	46.12	18.23	10.53	145.79	55.78	33.88	21.92	146.41	67.74	53.10	34.82
100×100	48 (432)	57.13	26.13	6.43	4.88	58.14	31.03	12.38	9.37	58.24	34.17	17.15	13.63
	56 (588)	84.91	36.22	7.96	5.62	79.32	40.93	15.98	12.08	86.95	46.37	23.04	18.07
	64 (768)	103.83	45.27	10.13	7.20	101.80	50.64	18.29	13.43	106.94	57.46	27.86	21.14
	72 (972)	125.10	56.48	11.52	7.45	124.98	60.49	20.74	14.76	124.09	65.42	30.02	22.81
$m = 6\,630$	80 (1200)	169.27	71.10	14.77	9.90	164.06	73.92	24.38	17.50	167.06	83.46	37.94	28.34
125×125	64 (768)	551.22	162.59	69.49	43.26	514.45	185.07	119.94	76.31	520.88	208.61	169.29	105.11
	72 (972)	621.59	180.18	67.90	44.14	653.11	222.98	142.39	90.24	670.56	257.52	209.77	127.82
	80 (1200)	832.09	227.59	85.07	54.36	807.29	258.31	158.36	99.36	794.50	293.82	229.15	140.61
	88 (1452)	964.25	257.64	91.14	57.96	998.12	301.44	173.33	107.80	1000.30	335.13	242.94	147.17
$m = 10\,293$	96 (1728)	1146.69	298.93	100.10	63.77	1093.40	336.32	193.62	121.44	1174.85	390.51	290.96	177.16
150×150	80 (1200)	806.53	266.77	72.39	53.64	817.47	310.92	151.02	110.97	772.51	335.17	205.54	147.73
	88 (1452)	963.90	325.02	91.13	69.15	986.68	366.39	167.29	123.76	1031.09	404.45	238.02	170.34
	96 (1728)	1204.41	369.11	91.46	69.72	1175.69	411.90	178.60	131.96	1179.82	454.06	257.34	182.61
	104 (2028)	1368.10	418.39	99.11	75.25	1322.91	462.84	198.73	147.31	1316.65	505.04	277.91	199.18
$m = 19\,244$	112 (2352)	1576.29	485.11	104.50	81.85	1705.01	550.06	222.14	161.60	1451.34	540.21	275.60	197.81
175×175	120 (2700)	5224.52	1277.55	427.88	266.43	5378.30	1350.53	821.44	446.03	5308.95	1560.84	1414.16	690.76
	128 (3072)	6030.39	1380.11	571.85	308.29	6052.93	1550.27	1035.63	527.83	5918.77	1616.89	1262.65	644.21
	136 (3468)	6816.26	1506.85	546.73	300.52	6737.70	1716.23	1142.69	576.39	6778.56	1830.07	1450.54	727.56
	144 (3888)	7475.43	1507.58	433.24	252.87	7403.45	1700.90	863.36	483.11	7444.04	1847.70	1201.31	650.11
$m = 23\,954$	152 (4332)	8337.66	1681.97	496.56	286.13	8298.14	1873.53	939.80	519.25	8357.65	2052.90	1328.87	721.88
200×200	208 (8112)	4992.42	1848.55	274.51	206.41	4877.93	1973.16	585.17	412.17	4584.01	1987.00	787.59	534.58
	216 (8748)	5268.11	1983.63	270.73	199.45	4884.62	2034.59	566.57	389.91	4798.81	2103.14	814.92	552.14
	224 (9408)	6184.71	2162.58	301.53	225.85	5491.03	2240.31	602.45	423.50	5120.73	2210.17	810.27	540.16
	232 (10092)	5481.62	2198.64	309.47	230.23	6379.35	2398.03	639.16	451.83	6109.89	2522.25	975.24	623.30
$m = 32\,612$	240 (10800)	5824.39	2268.39	341.82	237.01	6140.17	2443.76	688.94	449.38	6336.97	2603.77	1052.21	649.01

Table 6: Comparisons in terms of total CPU time (s) between "naïve" from-scratch evaluation and "naïve" update procedures (V1, V2 & V3) over 1000 iterations, where an iteration corresponds to a local change in the network, i.e. moving randomly a restricted set of buoys (1, 2 or 3). To achieve this, an initial network of N randomly distributed buoys is initialized and evaluated, then buoys are randomly moved. For a fixed number of moved buoys, at each iteration, the different methods evaluate the same network to ensure a fair comparison. Here using the **exponential** function.