

Supplementary document for  
**Studying parallel communication strategies  
with POSL**

Alejandro REYES-AMARO, Éric MONFROY, and Florian RICHOUX

LINA Inria–TASC, Université de Nantes, Nantes, France  
{alejandro.reyes, eric.monfroy,  
florian.richoux}@univ-nantes.fr

In order to make more comprehensible the presented abstract solvers, we will explain step-by-step one of them (the one presented in **Algorithm 9**).

```
1 abstract solver st_example
2   computation: M_S, M_V, M_SE, M_D;
3   connection: Ch_DP;
4 begin
5   execute(M_S,
6     while(Iter < k1)
7       execute(M_V,
8         execute(
9           ?(Sci < k2,
10            ?(Iter % k3,
11              min(M_SE, Ch_DP),
12              M_SE
13            ),
14            M_SE.Rand
15          ),
16          M_D
17        )
18      )
19    )
20  )
21 end
```

An abstract solver is declared in **Line 1** using the keyword **abstract solver**, and the keywords **computation** and **connection** to declare abstract computation modules (**Line 2**) and abstract connection modules (**Line 3**) respectively. The abstract solver code is written between the keywords **begin** (**Line 4**) and **end** (**Line 21**).

The abstract solver code:

- In **Line 5**, the operator **execute** executes the computation module  $M_S$  and uses its output as input to execute the module defined by the operator **while** starting in **Line 6**. The operator is defined from **Line 5** to **Line 20**.
- In **Line 6**, the operator **while** executes the module defined by the operator **execute** starting in **Line 7**, while the condition  $Iter < k1$  remains true. The operator is defined from **Line 6** to **Line 19**.
- In **Line 7**, the operator **execute** executes the computation module  $M_V$  and uses its output as input to execute the module defined by the operator **execute** starting in **Line 8**. The operator is defined from **Line 7** to **Line 18**.
- In **Line 8**, the operator **execute** executes the module defined by the operator **?** in **Line 9** and uses its output as input to execute the computation module  $M_D$  in **Line 16**. The operator is defined from **Line 8** to **Line 17**.
- In **Line 9**, the operator **?** executes the module defined by the operator **?** in **Line 10** if the condition  $Sci < k2$  remains true, or the instantiated computation module  $M_S.Rand$  in **Line 14** otherwise. The operator is defined from **Line 9** to **Line 15**.
- In **Line 10**, the operator **?** executes the module defined by the operator **min** in **Line 11** if the condition  $Iter \% k3$  remains true, or the computation module  $M_SE$  in **Line 12** otherwise. The operator is defined from **Line 10** to **Line 13**.
- In **Line 11**, the operator **min** executes both the computation module  $M_SE$  and the connection module  $Ch\_DP$  and returns the output with less cost.

## Appendix A: POSL strategies

**Algorithm 4:** Standard abstract solver

```
1 abstract solver st_std
2   computation: M_S, M_V, M_SE, M_D;
3 begin
4   execute(M_S, while(Iter < k1) // Iter : number of iterations
5     execute(M_V, execute(M_SE, M_D));
6   )
7 )
8 end
```

**Algorithm 5: Abstract solver combining two neighborhood functions using operator  $RHO$** 

```

1 abstract solver st_rho
2   computation: M_S, M_V1, M_V2, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1)
5       execute( rho(M_V1, M_V2), execute(M_SE, M_D)
6     )
7   )
8 end

```

**Algorithm 6: Abstract solver using the union of two neighborhood functions**

```

1 abstract solver st_u
2   computation: M_S, M_V1, M_V2, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1)
5       execute( union(M_V1, M_V2), execute(M_SE, M_D)
6     )
7   )
8 end

```

**Algorithm 7: Abstract solver to scape from local minima**

```

1 abstract solver st_eager
2   computation: M_S, M_V, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1) // Sci: number of iterations with the same cost
5       execute(M_V, execute(?(Sci < k2, M_SE, M_SE.Rand), M_D)
6     )
7   )
8 end

```

**Algorithm 8: Communicating abstract solver (sender solver)**

```

1 abstract solver st_eager_sender
2   computation: M_S, M_V, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1)
5       execute(M_V, execute(?(Sci < k2, send(M_SE), M_SE.Rand), M_D)
6     )
7   )
8 end

```

**Algorithm 9: Communicating abstract solver (receiver solver)**

```

1 abstract solver st_example
2   computation: M_S, M_V, M_SE, M_D;
3   connetion: Ch_DP;
4   begin
5     execute(M_S,
6       while(Iter < k1)
7         execute(M_V, execute(?(Sci < k2,?(Iter % k3, min(M_SE, Ch_DP), M_SE), M_SE.Rand), M_D)
8       )
9     )
10  end

```

**Algorithm 10: Reset-based abstract solver**

```

1 abstract solver st_hard
2   computation: M_S, M_R, M_V, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1)
5       execute(M_R, while(Iter < k2)
6         execute(M_V, execute(M_SE, M_D)
7       )
8     )
9   )
10  )
11 end

```

**Algorithm 11: Sender Reset-based abstract solver**

```

1 abstract solver st_hard_sender
2   computation: M_S, M_R, M_V, M_SE, M_D;
3   begin
4     execute(M_S, while(Iter < k1)
5       execute(M_R, while(Iter < k2)
6         execute(M_V, execute(M_SE, send(M_D)))
7       )
8     )
9   )
10  )
11 end

```

**Algorithm 12: Receiver Reset-based abstract solver A**

```

1 abstract solver st_hard_receiver_a
2   computation: M_S, M_R, M_V, M_SE, M_D;
3   connection: Ch_Sol;
4   begin
5     execute(M_S, while(Iter < k1)
6       execute(M_R, while(Iter < k2)
7         execute(M_V, execute(M_SE, min(M_D, Ch_Sol)))
8       end
9     )
10    end
11  )
12 end

```

**Algorithm 13: Receiver Reset-based abstract solver B**

```

1 abstract solver st_hard_receiver_b
2   computation: M_S, M_R, M_V, M_SE, M_D;
3   connection: Ch_Sol;
4   begin
5     execute(M_S, while(Iter < k1)
6       execute( ?(Iter % k2, M_R, min(M_R, Ch_Sol)),
7         while(Iter < k3)
8           execute(M_V, execute(M_SE, M_D))
9         end
10      )
11    end
12  )
13 end

```

**Algorithm 14: Receiver Reset-based abstract solver C**

```

1 abstract solver st_hard_receiver_c
2   computation: M_S, M_R, M_V, M_SE, M_D;
3   connection: Ch_Sol;
4   begin
5     execute(M_S, while(Iter < k1)
6       execute( min(M_R, Ch_Sol),
7         while(Iter < k2)
8           execute(M_V, execute(M_SE, M_D))
9         end
10      )
11    end
12  )
13 end

```

**Algorithm 15: Sender abstract solver using Tabu List**

```

1 abstract solver st_golomb_sender
2   computation: I, V, S, A, T;
3   begin
4     while(Iter < k1)
5       execute(I, execute( while(Iter % k2)
6         execute( execute(V, S), A)
7       end,
8       send(T)
9     )
10    )
11  end
12 end

```

**Algorithm 16: Receiver abstract solver using Tabu List**

```

1 abstract solver st_golomb_receiver
2   computation: I, V, S, A, T;
3   connection: Ch_Set;
4   begin
5     while(Iter < k1)
6       execute( execute(Ch_Set, I),
7         execute( while(Iter % k2)
8           execute( execute(V, S), A)
9         end,
10      T
11    )
12  )
13 end
14 end

```

## Appendix B: An example of a POSL's code

To illustrate what POSL code looks like, we present here the code used to solve the Social Golfers problem using the abstract solver presented in Algorithms 4, 5 and 6, with 50% of communicating solvers, sharing information *one to one*:

```
1 St_recv := cStrategy
2   oModule: OM_S, OM_V, OM_SE1, OM_SE2, OM_D;
3   {
4   {
5   <S>
6     OP.Cyc (BE.LoopBnd 50)
7       OM_S OP.|-> <S>
8         OP.Cyc ( BE.LoopBnd 5000 )
9           <S>
10            OM_V OP.|-> <S>
11              OP.? (BE.SCI 5)
12                <S>
13                  OP.? (BE.LoopBnd 10)
14                    OM_SE1
15                      <S> OP.Min OM_SE1 OCh (Ch1) </S>
16                        </S>
17                          OM_SE2
18                            </S> OP.|-> OM_D
19                      </S>
20                </S>
21          </S>
22    };

1 St_send := cStrategy
2   oModule: OM_S, OM_V, OM_SE1, OM_SE2, OM_D;
3   {
4   {
5   <S>
6     OP.Cyc ( BE.LoopBnd 50 )
7       OM_S OP.|-> <S>
8         OP.Cyc ( BE.LoopBnd 5000 )
9           OM_V OP.|-> <S>
10             OP.? (BE.SCI 5)
11               <S> OP.OSend (send_1) OM_SE1 </S> OP.|-> OM_D </S>
12               <S> OM_SE2 OP.|-> OM_D </S>
13             </S>
14     </S>
15 };

1 St_std := cStrategy
2   oModule: OM_S, OM_V, OM_SE1, OM_SE2, OM_D;
3   {
4   {
5   <S>
6     OP.Cyc (BE.LoopBnd 50)
7       OM_S OP.|-> <S>
8         OP.Cyc ( BE.LoopBnd 5000 )
9           OM_V OP.|-> <S>
10             OP.? (BE.SCI 5)
11               <S> OM_SE1 OP.|-> OM_D </S>
12               <S> OM_SE2 OP.|-> OM_D </S>
13             </S>
14     </S>
15 };
```

```
1 solver_sender := solver
2 {
3   cStrategy: St_send;
4   oModule: OM_S.PermBlocks, OM_V.AS_Golfers, OM_SE.FirstImpr, OM_SE.Rand, OM_D.AlwImpr;
5 };
6
7 solver_receiver := solver
8 {
9   cStrategy: St_recv;
10  oModule: OM_S.PermBlocks, OM_V.AS_Golfers, OM_SE.FirstImpr, OM_SE.Rand, OM_D.AlwImpr;
11  oChannel: OCh.DP_Last;
12 };
13
14 solver_single := solver
15 {
16   cStrategy: St_std;
17   oModule: OM_S.PermBlocks, OM_V.AS_Golfers, OM_SE.FirstImpr, OM_SE.Rand, OM_D.AlwImpr;
18 };
19
20 connections:
21
22 [solver_sender.send_1] OP.--> [solver_receiver.Ch1]10;
23 [solver_single]20;
```

Golfers

Instance	A000 / Simple Swap		A000 / AS swap		A000 / AS (rho) SS		A000 / AS U SS	
	Time	Iter.	Time	Iter.	Time	Iter.	Time	Iter.
<b>10-10-3</b>	62,160	108	980	274	65,470	244	64,750	192
40 cores	60,050	102	1,100	338	90,400	219	65,350	215
	60,900	104	400	126	22,760	99	55,930	225
	164,690	245	2,510	814	33,230	103	102,410	300
	57,950	115	2,380	866	20,860	90	112,620	298
	137,180	221	280	96	24,470	104	63,870	191
	139,970	233	320	103	23,710	119	3,670	84
	60,600	114	250	98	25,650	112	11,580	105
	73,670	120	2,760	986	27,540	94	63,830	200
	60,480	104	1,860	654	65,820	217	9,670	90
	64,240	124	690	285	28,980	98	58,400	219
	155,360	227	500	190	31,630	114	162,710	436
	56,860	105	560	187	69,000	232	182,930	397
	58,050	109	980	347	75,730	239	4,720	96
	148,500	236	320	118	80,710	240	130,950	333
	65,760	106	650	200	74,160	201	124,350	318
	56,050	111	240	106	95,450	357	9,260	98
	149,560	211	1,080	311	17,390	87	4,500	89
	65,510	101	1,420	483	16,010	75	3,500	88
	69,260	108	300	98	18,940	86	9,790	106
	50,830	102	240	86	33,360	106	132,650	325
	57,180	107	320	100	68,100	221	123,940	361
	54,430	100	270	98	23,190	114	8,410	109
	153,100	243	2,130	660	32,040	109	9,220	87
	59,940	110	1,070	332	16,390	93	58,940	220
	72,780	114	1,310	337	21,700	109	11,360	82
	154,310	229	2,180	754	20,190	99	10,700	91
	55,590	112	960	316	23,570	93	121,510	305
	63,300	112	1,900	598	21,640	110	8,780	84
	148,800	241	1,880	607	77,870	229	59,280	204
	<b>87,902</b>	<b>146</b>	<b>1,061</b>	<b>352</b>	<b>41,532</b>	<b>147</b>	<b>59,653</b>	<b>198</b>
	<b>41,961</b>	<b>58</b>	<b>791</b>	<b>268</b>	<b>26,001</b>	<b>72</b>	<b>55,019</b>	<b>110</b>

Golfers

Instance	B001 / Best Impr.		B001 / First Impr.	
	Time	Iter.	Time	Iter.
<b>5-3-7</b>	3,020	2,714	1,370	1,308
40 cores	3,860	3,433	1,380	1,308
	880	751	2,850	2,849
	15,040	13,458	1,010	958
	6,470	5,454	2,160	2,193
	2,160	1,878	450	444
	4,350	3,756	1,060	1,087
	3,080	2,908	2,680	2,677
	4,770	4,272	1,770	1,729
	3,170	2,842	480	504
	15,150	13,470	1,710	1,769
	1,910	1,690	780	784
	7,350	6,544	850	871
	3,190	2,895	440	482
	1,450	1,256	2,290	2,329
	1,680	1,541	1,640	1,549
	2,430	2,127	540	547
	4,640	4,021	420	439
	3,570	3,266	1,920	1,874
	19,450	17,174	2,020	1,979
	1,340	1,222	910	931
	5,100	4,559	1,510	1,517
	3,680	2,987	1,770	1,715
	3,680	2,987	1,970	2,013
	1,960	1,740	1,150	1,090
	1,970	1,712	630	664
	4,740	4,166	960	956
	8,780	7,878	600	622
	2,770	2,449	1,390	1,330
	8,340	7,475	1,120	1,130
	<b>4,999</b>	<b>4,421</b>	<b>1,328</b>	<b>1,322</b>
	<b>4,430</b>	<b>3,938</b>	<b>682</b>	<b>676</b>

Golfers

Instance	B001 / Best Impr.		B001 / First Impr.	
	Time	Iter.	Time	Iter.
<b>8-4-7</b>	6,300	1,192	1,720	424
40 cores	5,220	991	1,620	385
	3,130	573	2,990	718
	5,450	1,023	1,390	361
	2,690	502	4,700	1,104
	5,810	1,100	1,450	373
	3,250	623	1,170	300
	3,880	724	1,780	417
	6,270	1,163	1,220	308
	2,880	541	2,320	551
	1,520	279	1,660	406
	4,180	783	2,100	518
	5,460	980	880	220
	8,270	1,564	2,190	537
	4,240	797	1,470	361
	4,650	836	1,530	387
	8,230	1,548	1,580	380
	4,790	885	940	255
	4,920	907	4,110	962
	7,850	1,455	2,280	537
	6,340	1,197	1,600	390
	3,260	605	1,450	343
	3,470	653	2,090	498
	7,380	1,384	950	261
	5,170	973	1,630	405
	3,200	594	2,020	491
	5,500	1,046	1,660	401
	7,920	1,466	1,190	313
	5,570	1,035	1,330	324
	6,340	1,189	1,720	419
	<b>5,105</b>	<b>954</b>	<b>1,825</b>	<b>445</b>
	<b>1,776</b>	<b>334</b>	<b>840</b>	<b>191</b>



Golfers

Instance	B001 / Best Impr.		B001 / First Impr.	
	Time	Iter.	Time	Iter.
<b>9-4-8</b>	6,700	722	5,150	697
40 cores	10,500	1,136	9,710	1,312
	12,320	1,342	1,630	261
	8,250	891	7,580	923
	16,480	1,805	2,930	428
	11,080	1,193	5,100	693
	10,400	1,124	7,660	1,004
	12,670	1,385	2,810	420
	7,630	817	6,440	903
	20,590	2,250	3,990	558
	13,150	1,401	4,270	597
	11,360	1,227	5,410	729
	3,090	327	7,310	996
	7,210	773	3,700	534
	3,390	351	15,430	2,037
	19,630	2,136	1,280	223
	19,290	2,105	21,620	2,817
	8,090	874	2,470	381
	18,030	1,964	15,710	2,076
	15,600	1,712	2,130	314
	19,950	2,181	7,210	987
	14,200	1,538	7,360	985
	23,010	2,502	10,660	1,403
	19,090	2,043	3,250	453
	16,470	1,781	4,260	619
	9,870	1,075	10,550	1,403
	6,620	713	3,680	528
	11,440	1,242	4,330	598
	6,700	725	6,000	826
	8,410	917	3,540	488
	<b>12,374</b>	<b>1,342</b>	<b>6,439</b>	<b>873</b>
	<b>5,400</b>	<b>591</b>	<b>4,607</b>	<b>591</b>

Golfers

Instance	B001 / Best Impr.		B001 / First Impr.	
	Time	Itr.	Time	Itr.
<b>11-7-5</b>	8,650	565	1,340	179
40 cores	1,910	129	2,600	302
	6,440	447	2,490	280
	2,340	160	2,510	297
	2,370	160	3,460	350
	4,170	280	1,770	235
	6,770	469	3,720	402
	7,720	531	3,730	402
	4,270	283	2,520	325
	6,490	440	1,860	228
	2,920	194	1,300	202
	6,510	446	2,380	286
	6,510	446	1,390	216
	4,540	301	1,860	245
	6,320	433	3,630	387
	6,260	429	2,410	278
	5,170	339	1,950	265
	6,370	438	1,780	233
	6,580	427	1,660	255
	5,290	361	2,260	275
	6,480	446	2,690	323
	4,520	315	2,300	276
	4,670	317	2,280	276
	3,570	241	1,750	217
	3,010	193	1,800	245
	6,670	457	2,280	281
	4,840	327	2,270	281
	4,080	272	1,500	196
	5,330	348	1,600	265
	4,990	338	1,530	197
	<b>5,192</b>	<b>351</b>	<b>2,221</b>	<b>273</b>
	<b>1,671</b>	<b>114</b>	<b>692</b>	<b>58</b>

Golfers

Golfers

Instance				Cores		100% / -- >		100% / # - >	
						Time	Iter.	Time	Iter.
5	3	7	40	1,080	1,083	520	506		
Caption sender solver receiver solver receiver solver not tacking into account the received information				1,600	1,562	860	845		
				740	723	580	575		
				860	856	820	763		
				520	609	2,130	2,139		
				2,340	2,352	1,270	1,151		
				1,720	1,650	1,030	979		
				870	834	820	791		
				1,100	1,068	520	454		
				1,280	1,317	730	682		
				100	116	1,220	1,234		
				90	116	1,790	1,649		
				1,730	1,670	540	517		
				740	680	1,160	1,056		
				1,380	1,291	860	761		
				1,390	1,271	1,230	1,199		
				2,670	2,488	1,790	1,727		
				1,160	1,163	1,990	1,976		
				1,410	1,381	1,880	1,793		
				1,730	1,657	810	780		
				1,860	1,727	1,820	1,584		
				2,000	1,946	1,290	1,231		
				590	601	1,180	1,080		
				1,980	1,828	490	458		
				770	776	1,370	1,278		
				590	604	890	858		
				1,590	1,493	1,080	1,095		
				320	289	690	678		
				1,050	961	480	464		
580	568	1,740	1,700						
				1,195	1,156	1,119	1,067		
				646	608	499	484		
				53		80			

Golfers

Instance				Cores		100% / -- >		100% / # - >	
						Time	Iter.	Time	Iter.
8	4	7	40	1,250	287	1,500	358		
Caption sender solver receiver solver receiver solver not tacking into account the received information				1,390	327	1,450	362		
				1,320	321	1,990	455		
				2,220	522	1,010	253		
				850	213	2,130	466		
				2,200	499	1,010	247		
				1,040	271	820	178		
				2,860	667	1,040	258		
				2,710	630	1,370	309		
				320	97	2,250	532		
				410	102	1,930	431		
				1,140	283	950	209		
				870	215	1,580	356		
				960	254	1,210	277		
				710	184	2,620	620		
				2,150	468	1,470	339		
				800	207	1,160	304		
				2,720	642	1,720	441		
				490	152	2,470	587		
				460	134	1,770	393		
				1,340	335	970	234		
				1,220	292	490	148		
				2,040	496	740	197		
				680	175	1,050	256		
				1,590	382	2,210	502		
				1,790	435	2,440	562		
				1,030	257	930	220		
				660	160	1,370	336		
				1,040	283	920	221		
820	208	1,420	349						
				1,303	317	1,466	347		
				724	161	572	128		
				77		73			

Golfers

Instance				100% / -- >		100% / # - >	
9	4	8	40	Time	Iter.	Time	Iter.
				2,050	292	6,020	809
				4,200	594	4,010	573
			Caption	1,880	281	4,770	641
			sender solver	3,530	516	5,560	755
			receiver solver	1,000	175	5,250	686
			receiver solver not	5,960	793	6,180	832
			tacking into account the	3,730	532	15,050	2,001
			received information	2,610	362	5,660	752
				3,990	529	8,910	1,150
				4,760	633	8,610	1,098
				3,460	462	1,620	256
				9,290	1,187	4,480	608
				2,820	387	5,300	709
				10,050	1,357	1,030	166
				11,170	1,465	2,690	393
				4,200	565	5,970	803
				2,340	324	3,100	418
				4,140	561	10,100	1,311
				4,920	669	3,700	507
				2,810	412	800	147
				7,010	957	7,250	952
				10,160	1,313	3,790	507
				3,380	465	6,020	765
				1,960	315	4,230	583
				4,750	667	6,370	873
				2,760	365	2,360	330
				2,270	349	4,890	684
				6,480	856	3,050	403
				2,220	331	11,060	1,395
				1,510	202	7,560	975
				4,380	597	5,513	736
				2,720	347	3,066	389

Golfers

Instance				100% / -- >		100% / # - >	
11	7	5	40	Time	Iter.	Time	Iter.
Caption sender solver receiver solver receiver solver not tacking into account the received information				1,330	180	1,790	216
				2,240	223	1,530	207
				1,590	215	1,200	177
				2,090	216	1,580	217
				1,790	215	1,340	150
				1,590	216	1,530	166
				1,710	187	1,080	146
				1,010	141	2,570	286
				1,790	189	1,580	205
				2,370	221	1,530	196
				1,990	220	1,300	198
				1,990	220	1,540	203
				1,320	180	1,640	170
				1,890	249	1,330	194
				1,130	143	2,140	251
				1,250	150	1,720	189
				1,390	210	1,190	160
				1,210	159	1,910	203
				1,240	149	2,130	221
				1,830	204	1,900	243
				1,610	227	1,790	223
				1,710	226	1,430	173
				2,420	272	1,360	188
				1,820	240	1,360	185
				1,830	240	1,990	219
				2,220	270	1,400	208
				2,280	266	1,520	195
				1,680	192	2,270	253
				1,810	233	1,770	212
				2,730	352	1,340	191
				1,762	214	1,625	202
				420	45	348	31

Golfers



Golfers

Instance				Cores	50% / -- >		50% / # - >	
					Time	Iter.	Time	Iter.
5	3	7	40		870	848	690	735
Caption sender solver receiver solver receiver solver not tacking into account the received information					1,060	1,087	1,020	965
					580	580	840	844
					870	841	710	715
					480	484	570	557
					1,000	953	1,340	1,248
					2,460	2,458	310	282
					610	579	980	989
					1,610	1,559	210	209
					1,710	1,744	1,150	1,112
					1,280	1,300	900	921
					1,580	1,527	380	387
					1,100	1,090	1,620	1,549
					1,260	1,195	1,600	1,613
					780	746	1,000	1,025
					340	342	1,120	1,078
					1,210	1,156	2,360	2,364
					730	741	1,020	961
					1,140	1,042	960	961
					720	737	1,340	1,233
					1,240	1,189	740	738
					620	541	1,160	1,171
					1,300	1,205	280	277
					750	702	1,810	1,742
					830	826	2,000	2,037
					1,400	1,302	650	618
					250	227	2,020	2,050
					1,470	1,419	680	676
					1,000	1,036	1,040	1,002
	1,140	1,127	880	870				
				1,046	1,019	1,046	1,031	
				459	456	533	530	
				43		33		

Golfers

Instance				Cores	50% / -- >		50% / # - >	
					Time	Iter.	Time	Iter.
8	4	7	40	890	229	2,060	472	
Caption sender solver receiver solver receiver solver not tacking into account the received information				1,900	416	2,210	520	
				1,930	449	770	201	
				740	206	1,070	276	
				760	206	1,800	418	
				640	151	860	214	
				960	239	840	224	
				870	229	4,120	936	
				1,680	415	1,550	365	
				1,040	251	1,230	308	
				180	61	300	112	
				2,040	485	1,380	357	
				1,860	417	730	203	
				370	127	990	249	
				1,040	250	1,230	318	
				2,520	556	1,230	318	
				1,050	279	2,230	531	
				1,510	375	1,770	424	
				1,600	378	830	223	
				1,290	343	2,450	581	
				1,810	425	820	206	
				1,380	382	2,360	516	
				2,160	507	930	248	
				2,060	449	700	187	
				1,790	417	1,080	281	
				1,960	421	1,160	300	
				1,890	468	1,350	336	
				1,140	289	1,330	321	
				1,720	400	1,360	334	
1,210	297	2,420	596					
				1,400	337	1,439	353	
				579	122	768	167	
				43		27		

Golfers

Instance				Cores	50% / -- >		50% / # - >	
					Time	Iter.	Time	Iter.
9	4	8	40		4,590	640	8,980	1,143
Caption sender solver receiver solver receiver solver not tacking into account the received information					6,570	888	14,360	1,909
					5,530	774	7,770	999
					5,180	714	3,800	529
					3,360	477	1,950	282
					2,750	401	5,870	759
					4,050	565	5,570	755
					8,980	1,195	2,200	286
					8,920	1,161	7,180	970
					2,180	299	5,420	745
					6,570	925	1,580	255
					8,040	1,083	8,860	1,185
					2,970	415	6,110	846
					3,380	469	5,590	787
					4,390	550	3,300	442
					3,130	421	4,090	555
					2,840	405	3,930	572
					3,550	500	5,550	757
					2,450	369	4,540	631
					6,140	847	9,990	1,301
					1,660	240	11,020	1,427
					3,840	538	8,060	1,067
					3,350	478	4,590	647
					5,430	733	3,100	438
					3,090	456	2,050	316
					2,760	408	3,200	488
					5,830	764	4,210	593
					7,520	1,004	6,390	862
					8,510	1,103	3,040	424
	1,920	290	10,290	1,324				
				4,649	637	5,753	776	
				2,173	279	3,067	389	

Golfers

Instance				Cores	50% / -- >		50% / # - >	
11	7	5	40		Time	Iter.	Time	Iter.
					1,010	154	1,390	151
					1,960	220	1,810	238
				Caption	1,540	220	2,030	249
				sender solver	2,640	276	2,260	253
				receiver solver	1,970	230	2,030	276
				receiver solver not	1,640	209	2,080	248
				tacking into account the	2,160	195	1,540	213
				received information	1,640	183	1,510	181
					1,640	219	2,280	258
					2,290	257	1,390	176
					1,800	228	1,300	217
					1,800	218	1,640	179
					2,670	298	1,340	199
					1,640	206	1,400	200
					1,620	234	2,170	210
					2,160	232	1,940	259
					2,130	279	2,040	187
					1,900	237	1,900	206
					1,630	216	1,600	201
					1,840	197	1,280	153
					1,540	206	2,550	275
					2,290	248	2,500	301
					1,200	198	1,830	230
					1,410	187	1,360	186
					1,910	254	1,570	224
					1,680	211	2,660	288
					1,570	193	1,920	255
					1,050	145	1,890	196
					1,890	223	1,430	202
					2,250	234	2,020	252
					1,816	220	1,822	222
					401	33	395	39

Golfers

Golfers

Instance				Cores	25% / -- >		25% / # - >	
					Time	Iter.	Time	Iter.
5	3	7	40	1,820	1,745	510	518	
Caption sender solver receiver solver receiver solver not tacking into account the received information				980	905	1,330	1,266	
				610	547	990	994	
				2,030	1,884	890	844	
				410	419	1,350	1,264	
				330	321	380	363	
				1,710	1,675	1,430	1,305	
				390	382	820	807	
				1,560	1,451	1,090	1,079	
				790	774	2,180	2,072	
				520	450	730	727	
				470	481	1,100	1,163	
				1,350	1,370	810	849	
				550	563	830	819	
				820	776	2,040	2,005	
				580	587	910	871	
				280	312	990	926	
				450	444	1,440	1,388	
				1,310	1,282	690	643	
				570	542	2,740	2,622	
				1,210	1,251	960	904	
1,370	1,330	450	441					
1,200	1,187	680	708					
400	393	150	169					
260	272	2,820	2,805					
870	850	1,550	1,527					
1,610	1,616	610	682					
470	483	1,380	1,372					
1,260	1,195	2,380	2,385					
930	932	1,600	1,594					
				904	881	1,194	1,170	
				514	492	675	655	
				40		33		

Golfers

Instance				Cores	25% / -- >		25% / # - >	
					Time	Iter.	Time	Iter.
8	4	7	40	800	209	1,490	337	
Caption sender solver receiver solver receiver solver not tacking into account the received information				1,570	396	790	194	
				890	221	960	247	
				1,130	308	1,360	324	
				1,470	359	1,820	419	
				1,590	380	1,410	349	
				2,160	496	1,510	356	
				2,080	508	2,640	621	
				880	222	1,660	368	
				1,770	397	1,410	346	
				1,070	275	1,050	278	
				1,230	323	1,120	292	
				1,220	323	1,830	427	
				1,610	384	1,810	419	
				1,090	270	1,620	388	
				1,910	430	1,220	323	
				780	215	1,680	436	
				870	223	1,410	331	
				1,090	282	1,270	316	
				1,580	396	1,340	324	
				1,490	368	1,810	427	
				2,160	500	1,330	310	
				1,500	329	1,500	356	
				1,590	384	1,660	395	
				1,580	384	760	186	
				720	188	1,070	247	
				1,830	435	1,730	417	
				1,880	433	1,560	385	
				1,600	374	710	191	
820	203	2,450	565					
				1,399	341	1,466	352	
				435	94	434	96	
				30		27		

Golfers

Instance				Cores	25% / -- >		25% / # - >	
9	4	8	40		Time	Iter.	Time	Iter.
					3,280	472	7,040	923
					6,380	876	7,440	994
				Caption	6,570	882	1,920	315
				sender solver	3,150	405	2,180	348
				receiver solver	1,990	293	3,940	510
				receiver solver not	4,010	545	5,510	754
				tacking into account the	5,130	658	5,130	689
				received information	5,460	750	3,210	465
					6,360	856	4,050	565
					4,030	564	3,110	434
					8,970	1,235	5,160	714
					1,660	270	2,580	384
					1,710	252	2,360	359
					3,530	503	2,450	372
					4,130	588	4,370	602
					3,950	592	3,660	530
					1,630	272	7,130	991
					5,660	775	2,790	403
					7,110	914	9,110	1,178
					5,500	709	2,790	393
					3,870	541	2,550	390
					2,700	366	4,560	591
					3,290	450	6,390	873
					6,430	851	2,340	345
					7,860	1,083	5,560	766
					3,330	486	4,750	651
					2,840	412	6,480	843
					3,220	459	7,160	950
					3,700	512	7,700	1,021
					2,520	389	2,630	394
					4,332	599	4,535	625
					1,920	248	2,019	251



Golfers

Instance				Cores	25% / -- >		25% / # - >	
11	7	5	40		Time	Iter.	Time	Iter.
					2,330	268	2,180	250
					2,590	300	2,150	258
				Caption	2,600	300	1,760	237
				sender solver	2,600	300	1,760	260
				receiver solver	1,420	211	2,510	277
				receiver solver not	2,680	280	2,470	271
				tacking into account the	1,830	221	2,420	286
				received information	2,190	261	2,400	286
					2,220	268	1,840	220
					770	139	2,120	221
					1,280	170	1,710	218
					2,020	210	2,040	244
					1,520	207	2,050	244
					2,030	263	2,150	258
					2,550	269	1,760	241
					1,140	180	1,390	197
					2,530	280	1,540	196
					2,520	319	1,730	220
					1,530	201	2,200	246
					2,580	286	1,930	261
					1,560	161	1,090	179
					1,510	215	1,800	195
					1,980	270	1,200	179
					2,130	271	1,510	209
					1,640	211	1,810	229
					2,150	265	1,680	233
					1,260	150	1,830	250
					1,470	204	1,500	218
					2,730	320	2,070	225
					2,440	274	2,080	237
					1,993	242	1,630	224
					546	51	358	29

Costas Array

Instance	Cores	AS		St_a / 100% 1-1		St_a / 100% 1-n	
		Time	Iter.	Time	Iter.	Time	Iter.
17	40	1,810	19,861	860	9,481	1,000	13,702
		250	4,672	810	7,954	170	1,648
		120	1,234	830	7,870	350	4,172
		440	5,205	240	2,947	350	3,617
		1,570	16,639	180	2,252	210	2,593
		340	4,840	300	3,106	200	1,976
		450	4,831	280	3,978	160	1,617
		120	2,306	190	2,392	370	4,336
		1,020	10,638	200	1,868	1,040	16,596
		200	2,539	740	6,670	150	1,891
		1,160	14,293	380	6,395	230	2,566
		910	16,289	200	2,320	800	9,874
		550	6,652	700	9,247	920	9,599
		700	12,914	170	1,700	130	1,486
		280	2,884	1,030	10,403	780	8,361
		1,160	11,834	760	13,045	800	10,440
		850	14,665	760	7,660	340	4,647
		390	4,630	500	6,751	1,040	13,030
		890	9,382	160	1,360	380	4,243
		560	9,262	140	1,528	360	3,763
		1,580	16,999	120	1,198	500	6,985
		1,620	29,008	380	4,804	110	1,312
		310	5,462	220	1,981	100	1,066
		280	2,974	180	2,833	150	2,560
		180	3,367	190	1,928	560	5,119
		380	3,631	980	8,974	590	9,214
		270	3,064	1,130	10,750	650	7,919
		1,000	10,571	260	2,572	160	1,922
		1,280	14,581	380	3,324	260	2,638
		230	4,099	120	1,465	80	1,035
		100	1,752	40	689	540	11,224
		560	6,574	790	13,384	760	13,072
		1,150	12,022	80	844	370	5,209
		720	10,345	610	5,650	110	2,298
		1,210	13,006	470	4,748	120	1,182
		280	3,023	100	1,036	420	4,527
		110	1,195	50	562	100	1,069
		990	10,448	480	7,780	80	1,054
		910	9,787	510	4,883	690	7,756
		1,020	18,244	630	11,797	80	871
		920	10,564	610	6,125	1,100	18,772
		730	10,939	260	2,734	80	872
		780	12,232	270	3,322	690	7,912
		1,430	24,616	240	2,500	910	14,065
		900	15,970	210	2,041	420	7,740
		727	9,557	416	4,819	431	5,723
		469	6,439	296	3,589	317	4,744

Costas Array

Instance	Cores	St_a / 50% 1-1		St_a / 50% 1-n	
		Time	Iter.	Time	Iter.
17	40	330	3,964	760	12,946
		550	10,750	140	1,963
		840	15,835	120	1,384
		280	3,019	780	9,553
		960	11,142	720	12,013
		1,030	15,841	20	172
		340	5,251	570	7,855
		330	4,153	110	1,259
		260	3,313	290	3,496
		290	4,834	1,270	22,888
		10	119	1,020	20,027
		170	3,730	1,430	16,756
		370	8,019	460	6,115
		1,640	23,068	290	5,632
		1,060	20,573	40	571
		360	5,329	1,700	23,350
		410	5,146	430	8,873
		580	7,987	420	5,695
		810	9,737	990	11,596
		90	1,112	250	3,478
		220	2,644	210	2,500
		530	10,198	300	3,691
		1,310	13,846	490	7,138
		70	1,348	760	7,840
		570	6,703	440	7,016
		550	6,703	1,540	30,901
		20	427	1,640	19,264
		300	3,779	1,380	27,048
		180	2,364	180	3,737
		750	8,929	720	8,287
		630	9,292	1,430	24,226
		50	681	660	6,959
		600	7,396	160	2,191
		1,330	15,478	470	9,328
		780	14,206	40	402
		790	12,373	350	4,260
		100	1,171	490	10,333
		340	4,189	960	11,074
		260	3,493	60	616
		620	8,236	30	328
		510	5,246	380	5,173
		660	6,893	220	2,638
		20	241	480	6,091
		1,330	21,556	960	12,664
		1,920	39,955	320	5,176
		559	8,228	588	8,767
		443	7,556	477	7,774

Costas Array

Instance	Cores	St_b / 100% 1-1		St_b / 100% 1-n	
		Time	Iter.	Time	Iter.
17	40	10	322	2,420	32,411
		320	5,560	1,690	17,034
		790	9,226	650	6,397
		40	541	1,200	22,472
		1,280	20,420	640	6,389
		1,240	12,358	610	5,670
		850	10,123	920	9,427
		280	4,840	250	3,058
		880	10,507	960	16,715
		430	5,125	940	9,898
		60	625	100	1,243
		340	6,875	870	17,608
		60	583	180	1,885
		980	11,539	790	7,738
		90	949	470	5,982
		230	2,281	130	1,366
		830	12,990	120	1,780
		550	6,721	170	1,932
		330	6,537	40	432
		320	3,317	320	3,547
		760	11,194	200	3,073
		420	4,405	80	1,360
		160	1,955	20	130
		90	1,105	430	7,915
		200	3,649	30	373
		300	3,982	210	2,869
		160	1,748	50	544
		790	8,998	470	5,890
		280	3,715	370	3,932
		120	1,318	170	2,164
		1,370	14,143	90	1,546
		390	4,216	1,160	14,086
		630	7,579	120	1,447
		170	1,990	120	1,186
		90	1,012	300	5,012
		1,590	16,567	300	3,133
		320	3,976	20	280
		1,310	17,230	470	6,574
		120	1,567	260	2,423
		190	3,301	90	1,048
210	2,143	360	3,436		
470	9,954	600	7,060		
360	5,307	290	3,217		
160	1,691	380	5,341		
1,020	17,737	280	2,581		
		480	6,265	452	5,769
		417	5,321	477	6,578

Costas Array

Instance	Cores	St_b / 50% 1-1		St_b / 50% 1-n	
		Time	Iter.	Time	Iter.
17	40	990	13,492	220	4,852
		130	1,654	2,190	27,028
		960	19,321	810	13,510
		570	12,136	80	814
		230	4,858	680	12,859
		740	8,590	630	6,412
		410	5,498	120	1,390
		320	4,429	440	4,309
		300	4,039	490	5,614
		320	4,123	450	9,082
		1,180	15,841	110	1,252
		420	7,598	2,330	42,391
		50	616	140	1,780
		1,620	18,894	120	1,557
		1,890	26,893	500	5,965
		130	1,924	1,310	27,804
		1,280	13,420	500	7,017
		160	1,861	430	5,332
		10	211	290	3,736
		110	1,369	80	859
		120	1,573	160	3,142
		100	872	40	507
		60	691	100	1,219
		1,050	14,983	580	6,859
		1,670	34,654	100	1,486
		490	6,250	170	2,440
		1,560	26,260	200	2,719
		10	187	40	382
		130	1,246	430	6,370
		300	4,066	630	7,480
		290	3,562	360	4,621
		1,030	13,999	1,500	17,857
		500	10,426	1,240	16,702
		300	3,451	660	9,184
		110	1,441	160	1,642
		970	20,263	200	2,668
		40	988	170	2,218
		320	6,691	260	2,918
		480	10,150	620	13,171
		640	6,592	1,030	19,681
		20	451	90	1,036
		90	1,609	1,100	13,264
		480	8,611	440	4,471
		270	3,601	280	3,415
		940	15,943	210	2,746
		529	8,118	504	7,372
		506	8,192	524	8,544

Costas Array

Instance	Cores	St_c / 100% 1-1		St_c / 100% 1-n	
		Time	Iter.	Time	Iter.
17	40	40	340	960	18,532
		200	1,936	980	10,132
		180	1,966	290	3,196
		1,170	11,605	540	5,428
		520	10,654	1,300	26,122
		390	5,275	180	2,005
		380	4,012	140	2,065
		660	6,797	170	1,849
		340	4,321	180	2,233
		240	2,527	400	4,198
		320	3,799	680	7,546
		310	3,484	780	7,377
		400	4,138	890	11,849
		1,100	21,040	870	9,637
		1,110	19,507	110	1,406
		240	3,493	120	1,315
		1,250	25,261	250	2,574
		320	6,529	660	9,044
		360	4,054	200	2,683
		340	3,547	630	12,604
		1,070	10,996	630	6,409
		320	3,574	570	5,563
		340	7,062	800	7,642
		360	4,459	500	10,312
		1,100	19,301	500	8,659
		160	1,672	590	6,070
		1,410	16,363	130	1,423
		150	1,615	1,820	30,643
		300	3,160	830	16,030
		1,310	15,066	720	8,838
		1,360	25,852	740	8,437
		590	12,271	100	1,027
		230	2,788	600	7,387
		10	313	580	6,493
		390	8,149	170	1,957
		390	6,727	150	1,771
		370	7,654	340	4,627
		20	250	570	6,751
		120	1,518	500	5,932
		100	1,201	20	265
		170	2,237	280	3,910
		230	2,767	250	2,785
		490	6,028	190	2,368
		480	8,626	180	3,091
		920	9,349	450	5,035
		495	7,184	501	6,783
		400	6,632	359	6,218

Costas Array

Instance	Cores	St_c / 50% 1-1		St_c / 50% 1-n	
		Time	Iter.	Time	Iter.
17	40	1,990	24,955	70	793
		40	551	360	5,321
		720	15,226	220	2,374
		730	7,378	210	2,416
		1,200	14,257	540	7,306
		720	9,265	40	583
		870	10,972	410	4,288
		270	5,617	870	12,076
		190	2,620	580	11,218
		1,010	19,234	720	8,242
		1,280	15,586	1,600	30,455
		330	4,636	1,600	15,751
		870	11,847	180	2,164
		300	4,129	2,080	25,051
		80	1,066	1,350	18,460
		410	5,167	1,380	13,921
		1,570	29,595	440	4,351
		860	9,987	380	4,663
		80	1,765	370	5,008
		390	3,992	230	4,882
		310	3,754	320	3,643
		960	10,719	130	2,618
		970	11,053	400	4,690
		910	17,042	230	2,917
		1,150	15,991	400	4,807
		320	3,959	110	1,393
		120	2,460	260	3,580
		520	8,194	1,280	25,900
		520	5,842	400	5,084
		590	12,658	300	4,077
		220	3,193	290	2,944
		100	1,184	200	2,211
		10	241	210	2,983
		700	8,263	220	2,239
		390	5,312	310	3,288
		720	11,386	770	7,825
		160	2,080	1,790	22,204
		640	12,271	180	1,978
		870	10,204	1,710	36,013
		530	5,959	470	4,819
		390	8,155	660	12,346
		160	1,985	910	10,369
		810	10,024	920	11,116
		180	3,850	440	4,474
		280	3,375	420	5,179
		588	8,378	599	8,178
		432	6,437	525	8,289

Golomb Ruler

**Golomb Ruler 8-34**

Using Tabu List (no communication)

Without Tabu List (no communication)

Eps = 4

Norm = 8

tabu size = 15

time	iterations	restarts
110	743	3
100	768	3
50	518	2
20	88	0
20	202	1
30	233	1
20	197	0
10	112	0
20	86	0
40	229	1
10	88	0
20	88	0
20	89	0
100	970	4
30	173	0
40	401	2
20	169	0
30	173	0
30	196	0
170	1543	7
90	576	2
60	602	3
60	570	2
60	547	2
10	86	0
10	138	0
20	142	0
10	87	0
50	487	2
20	167	0
<b>43</b>	<b>349</b>	<b>1</b>
38	334	2

time	iterations	restarts
70	685	3
50	432	2
50	400	1
90	801	4
50	371	1
60	603	3
110	1118	5
30	286	1
40	378	1
30	315	1
30	259	1
50	431	2
20	204	1
130	1290	6
100	961	4
60	604	3
20	263	1
20	194	0
10	112	0
30	290	1
20	166	0
40	343	1
10	114	0
140	1200	5
30	259	1
20	203	1
40	260	1
10	88	0
20	202	1
40	259	1
<b>47</b>	<b>436</b>	<b>2</b>
35	330	2



Golomb Ruler

**Golomb Ruler 10-55**

Using Tabu List (no communication)

Without Tabu List (no communication)

Eps = 4

Norm = 8

tabu size = 15

time	iterations	restarts
4740	20069	13
13990	56986	37
7980	33958	22
6810	28162	18
9300	37336	24
21410	89951	59
3340	13506	9
3190	13458	8
1230	4449	2
2320	9792	6
5520	23529	15
4330	17268	11
3170	12892	8
1050	3734	2
8090	34830	23
5310	21741	14
2950	12597	8
620	2892	1
11370	48732	32
2790	11731	7
2400	9733	6
2110	8268	5
11010	47398	31
1560	6140	4
3270	13937	9
640	2355	1
990	3698	2
480	1768	1
3120	12896	8
2600	11322	7
<b>4,923</b>	<b>20,504</b>	<b>13</b>
4,687	19,743	13

time	iterations	restarts
5360	21072	14
1700	7295	4
6590	27135	18
900	3466	2
5710	24439	16
4000	16161	10
4760	21600	14
1480	5834	3
15100	63697	42
12840	54234	36
4270	19252	12
7360	32230	21
7330	32695	21
2370	10163	6
6890	29354	19
1080	4570	3
10340	44978	29
13190	56969	37
8370	36197	24
3260	14755	9
970	4450	2
920	4247	2
4250	17070	11
3630	15300	10
8690	36498	24
2000	7965	5
3760	15501	10
7390	29296	19
3370	14731	9
1430	6169	4
<b>5,310</b>	<b>22,577</b>	<b>15</b>
3,863	16,488	11

Golomb Ruler

**Golomb Ruler 11-72**

Using Tabu List (no communication)

Without Tabu List (no communication)

Eps = 4

Norm = 8

tabu size = 15

time	iterations	restarts
32170	58451	19
86650	156935	52
161460	293271	97
220280	398176	132
74050	136241	45
66420	119174	39
179500	327108	109
117350	212795	70
103960	192214	64
9800	17999	5
25290	45418	15
108060	201626	67
119020	219628	73
18550	34456	11
210190	384418	128
53520	100043	33
127630	230691	76
52620	96214	32
214640	383587	127
28080	50999	16
79130	146484	48
55850	103242	34
91340	167080	55
7270	12421	4
184280	338482	112
1620	2999	0
12520	22971	7
3640	6928	2
34220	64652	21
71590	132829	44
<b>85,023</b>	<b>155,251</b>	<b>51</b>
67,223	121,929	41

time	iterations	restarts
273700	508450	169
30790	56778	18
103280	192006	64
53230	95905	31
41540	78212	26
58890	104483	34
90190	167372	55
133910	248677	82
82350	152068	50
150320	273317	91
1430	2785	0
95300	173065	57
173490	311376	103
51820	95176	31
21590	40349	13
40000	73144	24
123830	232758	77
82010	149688	49
60880	113789	37
98000	181972	60
56980	104485	34
129980	236088	78
98100	180725	60
78750	146276	48
166690	302689	100
82660	151445	50
92160	164589	54
142440	263995	87
54130	98081	32
24350	43139	14
<b>89,760</b>	<b>164,763</b>	<b>54</b>
55,859	102,931	34

Golomb Ruler

**Golomb Ruler 8-34**

Using Tabu List (communication 1-1)

Using Tabu List (communication 1-n)

Eps = 4

Norm = 8

tabu size = 40

time	iterations	restarts
40	431	2
30	199	0
30	140	0
30	316	1
80	569	2
60	340	1
40	202	1
30	140	0
30	113	0
160	1202	6
90	570	2
60	430	2
30	262	1
50	395	1
30	143	0
40	337	1
30	263	1
40	312	1
20	193	0
20	120	0
30	286	1
30	112	0
20	174	0
10	89	0
80	573	2
20	60	0
30	289	1
70	460	2
10	4	0
90	543	2
<b>44</b>	<b>309</b>	<b>1</b>
31	233	1

time	iterations	restarts
20	87	0
10	30	0
20	113	0
80	543	2
70	403	2
10	57	0
70	460	2
40	203	1
10	121	0
10	62	0
80	594	2
40	203	1
30	203	1
10	87	0
70	342	1
90	630	3
40	368	1
10	3	0
80	632	3
60	488	2
20	203	1
10	30	0
70	492	2
40	170	0
80	570	2
40	261	1
100	768	3
10	60	0
50	289	1
10	3	0
<b>43</b>	<b>283</b>	<b>1</b>
30	225	1

Golomb Ruler

**Golomb Ruler 10-55**

Using Tabu List (communication 1-1)

Using Tabu List (communication 1-n)

Eps = 4

Norm = 8

tabu size = 40

time	iterations	restarts	time	iterations	restarts
1370	5957	3	8180	35159	23
440	1613	1	7920	33205	22
6440	26337	17	1350	5190	3
590	2402	1	3030	11119	7
310	1258	0	3460	15232	10
1790	7891	5	630	2326	1
2880	11075	7	9300	36107	24
1700	6537	4	1680	6760	4
260	797	0	170	757	0
550	2409	1	5520	21400	14
5750	21993	14	210	758	0
6800	26270	17	1100	4314	2
9100	35963	23	8150	32458	21
1070	4018	2	1510	5813	3
3010	11320	7	460	1897	1
4590	17455	11	2230	8359	5
10120	40753	27	3650	14376	9
3620	14283	9	1240	4771	3
3080	12199	8	580	2432	1
430	1674	1	460	1743	1
9580	39655	26	5560	22256	14
3060	11673	7	1470	5451	3
7210	28998	19	960	3858	2
1310	5008	3	1280	5100	3
6620	27298	18	4800	20616	13
8850	35597	23	3760	14826	9
2480	10118	6	1620	6434	4
9170	32662	21	1370	5199	3
1460	5952	3	7910	29852	19
3460	13935	9	5290	20388	13
<b>3,903</b>	<b>15,437</b>	<b>10</b>	<b>3,162</b>	<b>12,605</b>	<b>8</b>
3,225	12,789	9	2,824	11,405	8

Golomb Ruler

**Golomb Ruler 11-72**

Using Tabu List (communication 1-1)

Using Tabu List (communication 1-n)

Eps = 4

Norm = 8

tabu size = 40

time	iterations	restarts
58150	104687	34
211410	387838	129
5580	10135	3
81170	151964	50
44770	80806	26
92430	163137	54
62460	111108	37
48680	87420	29
50410	89278	29
192780	356580	118
84510	158490	52
174050	321314	107
62130	110587	36
116400	209895	69
69700	127352	42
78790	143793	47
53510	100957	33
42740	76143	25
73150	131271	43
38160	72321	24
12310	22243	7
96690	179268	59
190280	350380	116
78940	143379	47
98090	180525	60
159820	295856	98
96030	175445	58
78610	141931	47
24700	44064	14
86470	158172	52
<b>85,431</b>	<b>156,211</b>	<b>52</b>
52,610	97,330	32

time	iterations	restarts
170270	312109	104
62810	111833	37
14140	26681	8
53730	97756	32
16230	28758	9
24030	42519	14
39870	74069	24
59800	107992	35
16660	30007	10
79380	146171	48
95520	175044	58
118760	217352	72
21140	37044	12
19580	34345	11
124340	224683	74
85260	152680	50
10690	19041	6
75000	136039	45
26580	46451	15
154340	287165	95
7170	12523	4
44110	81521	27
90380	168320	56
32580	59889	19
47430	83487	27
114890	215689	71
47620	85957	28
13730	24413	8
77170	144113	48
67510	125686	41
<b>60,357</b>	<b>110,311</b>	<b>36</b>
43,951	81,296	27

Golomb Ruler

**Golomb Ruler 8-34**

Sequential (without tabu list)

Sequential (with tabu list)

Eps = 4

Norm = 8

tabu size = 15

time	iterations	restarts
80	1232	6
950	15915	79
400	6803	34
600	10316	51
880	14802	74
2050	33658	168
2580	43742	218
200	3313	16
690	10887	54
1640	26832	134
1040	17171	85
210	3486	17
240	4082	20
100	1717	8
130	2483	12
1040	18059	90
1720	28799	143
1070	18138	90
1290	21230	106
40	740	3
460	7604	38
100	1741	8
940	15030	75
110	1887	9
1760	28857	144
420	7172	35
1440	25339	126
780	13258	66
220	3857	19
670	11030	55
<b>795</b>	<b>13,306</b>	<b>66</b>
668	11,154	56

time	iterations	restarts
1090	17140	85
10	233	1
240	3860	19
2190	36030	180
2350	37203	186
700	11366	56
150	2634	13
740	11461	57
870	13795	68
80	1260	6
50	994	4
450	7202	36
970	15796	78
510	8142	40
100	1547	7
1230	20487	102
500	8287	41
560	9311	46
860	14089	70
70	1345	6
510	8343	41
450	7342	36
80	1401	7
260	4340	21
1960	31257	156
260	4086	20
1340	21514	107
250	3942	19
30	546	2
1070	17399	86
<b>664</b>	<b>10,745</b>	<b>53</b>
639	10,260	51

Golomb Ruler

**Golomb Ruler 10-55**

Sequential (without tabu list)

Sequential (with tabu list)

Eps = 4  
 Norm = 8  
 tabu size = 15

time	iterations	restarts
11540	85707	57
106120	759399	506
58220	411362	274
49070	342733	228
20200	146027	97
47620	327598	218
55600	372135	248
90620	610539	407
7650	52896	35
145150	966493	644
33210	224551	149
96990	662726	441
47300	318721	212
15960	108936	72
18560	130505	87
7480	50949	33
68200	471494	314
92600	620422	413
22950	156637	104
45450	306110	204
34570	188025	125
112110	768339	512
142360	965061	643
142300	960895	640
78720	528396	352
56810	389701	259
12120	82335	54
58860	387600	258
208790	1417178	944
106150	729103	486
<b>66,443</b>	<b>451,419</b>	<b>301</b>
49,569	336,859	225

time	iterations	restarts
17720	117993	78
146880	993766	662
63660	428626	285
10090	67926	45
75560	514704	343
53700	372893	248
147140	1011234	674
57570	400503	267
44530	295743	197
40970	277320	184
410	2791	1
40840	271494	180
57140	382793	255
38570	256086	170
134890	894727	596
91410	608716	405
57340	389626	259
105230	728128	485
114190	790798	527
110290	758168	505
23310	157930	105
14320	88766	59
151240	1007429	671
88750	600499	400
39030	257327	171
51060	221101	147
37810	198660	132
24270	156268	104
5610	32591	21
193370	1122786	748
<b>67,897</b>	<b>446,913</b>	<b>297</b>
50,024	328,912	219

Golomb Ruler

**Golomb Ruler 11-72**

Sequential (without tabu list)

Sequential (with tabu list)

Eps = 4

Norm = 8

tabu size = 15

time	iterations	restarts
279920	536214	178
123870	246321	82
158860	486633	162
59290	187837	62
298830	914400	304
216040	673087	224
98300	301330	100
47670	107164	35

time	iterations	restarts
26420	80481	26
126880	414007	138
140600	462828	154
68180	214270	71
272440	859895	286
10300	34398	11
124410	417131	139
72180	243698	81
216000	716845	238

<b>160,348</b>	<b>431,623</b>	<b>143</b>
96,116	272,911	91

<b>117,490</b>	<b>382,617</b>	<b>127</b>
85,624	275,748	92