



Universidade Federal de Minas Gerais

Programa de Pós-Graduação em Engenharia Elétrica da UFMG

Mestrado e Doutorado em Engenharia Elétrica

TRABALHO 1

OTIMIZAÇÃO EM ENGENHARIA

MÉTODO DE PONTOS INTERIORES CONJUGADO COM O SIMPLEX PARA A RESOLUÇÃO DO PROBLEMA KLEE-MINTY

VINICIUS CLAUDINO FERRAZ

BELO HORIZONTE

2021/2

Sumário

1	Resultados dos algoritmos de solução de problemas de programação linear em relação à variação da dimensão do problema	3
1.1	Tempo de execução utilizando os comandos tic e toc do Matlab (s)	3
1.2	Número de iterações do algoritmo	3
1.3	Gráficos Tempo \times Dimensão	4
1.4	Gráficos Iterações \times Dimensão	5
1.5	Erro da função objetivo (%)	6
1.6	Erro da solução (%)	6
1.7	Conclusão	7
2	Avaliar o comportamento dos algoritmos em função da dimensão do problema de Klee-Minty sob o ponto de vista do passo	7
2.1	Pontos Interiores	7
2.1.1	Tempo de execução utilizando os comandos tic e toc do Matlab (s) — PI	7
2.1.2	Número de iterações do algoritmo — PI	8
2.1.3	Gráficos Dimensão \times Tempo — PI	8
2.1.4	Gráficos Dimensão \times Iteração — PI	9
2.1.5	Erro da função objetivo (%) — PI	10
2.1.6	Erro da solução (%) — PI	10
2.2	Híbrido	10
2.2.1	Tempo de execução utilizando os comandos tic e toc do Matlab (s) — Híbrido	10
2.2.2	Número de iterações do algoritmo — Híbrido	11
2.2.3	Gráficos Dimensão \times Tempo — Híbrido	11
2.2.4	Gráficos Dimensão \times Iteração — Híbrido	12
2.3	Conclusão	13
3	Avaliar o esforço computacional quanto a variação do gap de dualidade para os algoritmos PI e Híbrido	15
3.1	Pontos Interiores	15
3.1.1	Tempo de execução utilizando os comandos tic e toc do Matlab (s) — PI	15
3.1.2	Número de iterações do algoritmo — PI	15
3.1.3	Gráficos Dimensão \times Tempo — PI	16
3.1.4	Gráficos Dimensão \times Iteração — PI	16
3.1.5	Erro da função objetivo (%) — PI	17
3.1.6	Erro da solução (%) — PI	17
3.2	Híbrido	17
3.2.1	Tempo de execução utilizando os comandos tic e toc do Matlab (s) — Híbrido	17
3.2.2	Número de iterações do algoritmo — Híbrido	17
3.2.3	Gráficos Dimensão \times Tempo — Híbrido	18
3.2.4	Gráficos Dimensão \times Iteração — Híbrido	18
3.3	Conclusão	20
4	Resultados relacionados ao esforço computacional para o algoritmo Híbrido (Detalhamento dos métodos de Pontos Interiores e SIMPLEX)	21
5	Ponto Inicial e Parâmetros do Problema de Klee-Minty	22
6	Conclusão	23

1. Resultados dos algoritmos de solução de problemas de programação linear em relação à variação da dimensão do problema

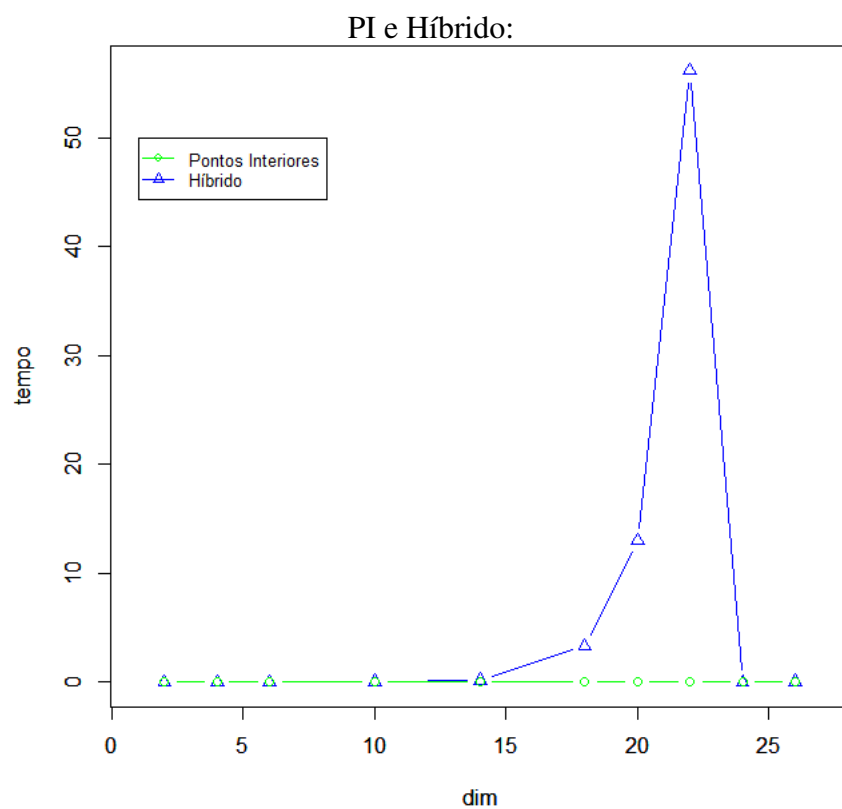
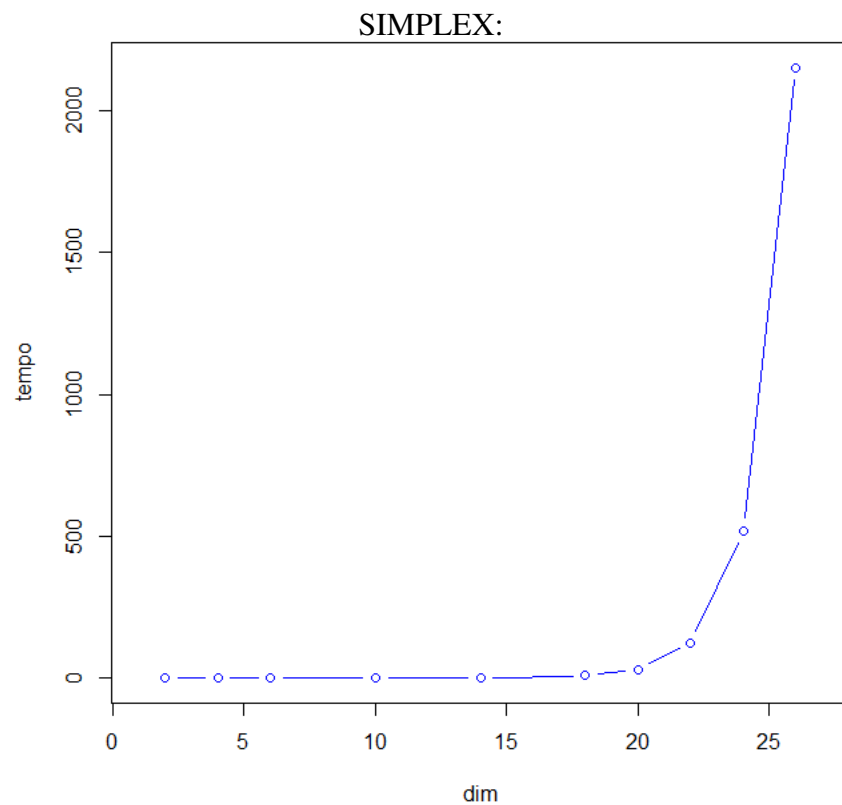
1.1 Tempo de execução utilizando os comandos tic e toc do Matlab (s)

Dimensão	SIMPLEX	PI	Híbrido
2	0.002625	0.003178	0.012878
4	0.004228	0.001685	0.003381
6	0.004462	0.002101	0.004666
10	0.095060	0.003326	0.021203
14	0.678472	0.007566	0.175043
18	8.577663	0.008210	3.281314
20	30.861928	0.010404	12.975993
22	124.140249	0.010937	56.196515
24	518.626911	0.012752	0.013291
26	2151.667869	0.012549	0.022114

1.2 Número de iterações do algoritmo

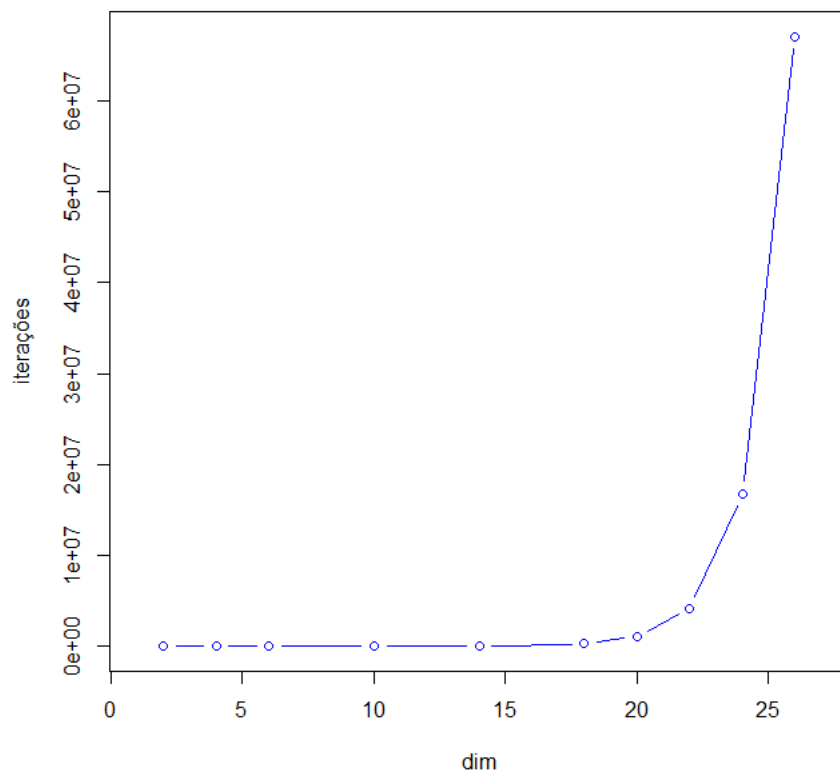
Dimensão	SIMPLEX	PI	Híbrido
2	3	2	1
4	15	2	4
6	63	2	13
10	1023	2	205
14	16383	2	3277
18	262143	2	52429
20	1048575	2	209716
22	4194303	2	838861
24	16777215	2	18
26	67108863	2	12

1.3 Gráficos Tempo \times Dimensão

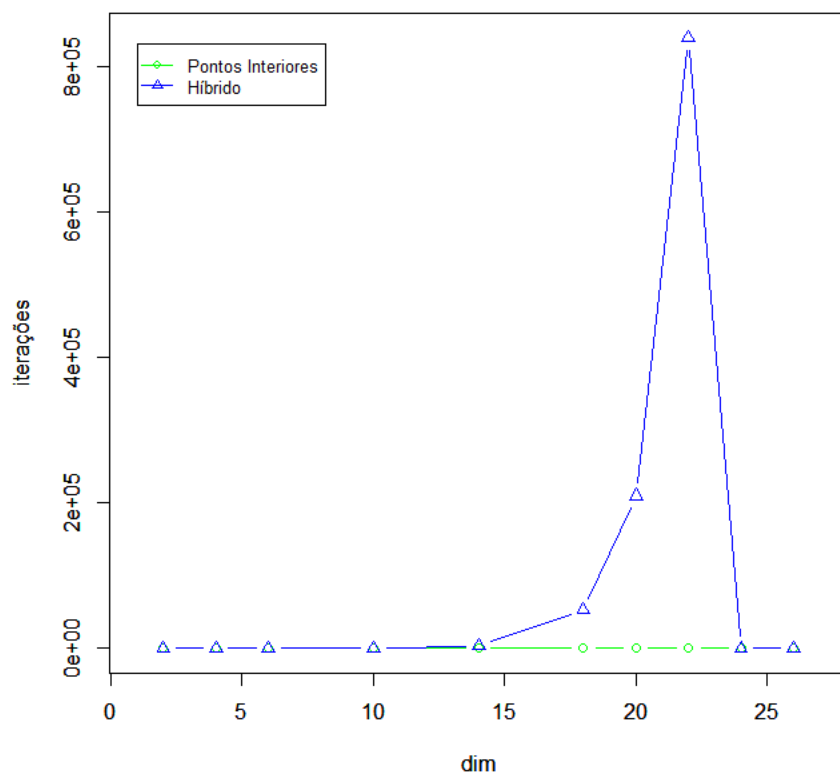


1.4 Gráficos Iterações \times Dimensão

SIMPLEX:



PI e Híbrido:



1.5 Erro da função objetivo (%)

Dimensão	SIMPLEX	PI	Híbrido
2	0	20	0
4	0	24.8	0
6	0	24.992	0
10	0	24.999987	0
14	0	25	0
18	0	25	0
20	0	25	0
22	0	25	0
24	0	25	0
26	0	25	0

1.6 Erro da solução (%)

Dimensão	SIMPLEX	PI	Híbrido
2	0	10	0
4	0	10.050373	0
6	0	10.050378	0
10	0	10.050378	0
14	0	10.050378	0
18	0	10.050378	0
20	0	10.050378	0
22	0	10.050378	0
24	0	10.050378	0
26	0	10.050378	0

1.7 Conclusão

O SIMPLEX consegue resolver o problema com erro zero, com gigante deficiência no desempenho e no número de iterações.

O método de pontos interiores tem deficiência na margem de erro, mas é o campeão de desempenho e número de iterações.

O Híbrido, como roda ambos, é exato, e tem número de iterações intermediário. Raramente esse número é alto, e a gente imagina que são casos patológicos, uma vez que o desempenho foi de centésimos de segundo em dimensão maior ou igual a 24.

2. Avaliar o comportamento dos algoritmos em função da dimensão do problema de Klee-Minty sob o ponto de vista do passo

2.1 Pontos Interiores

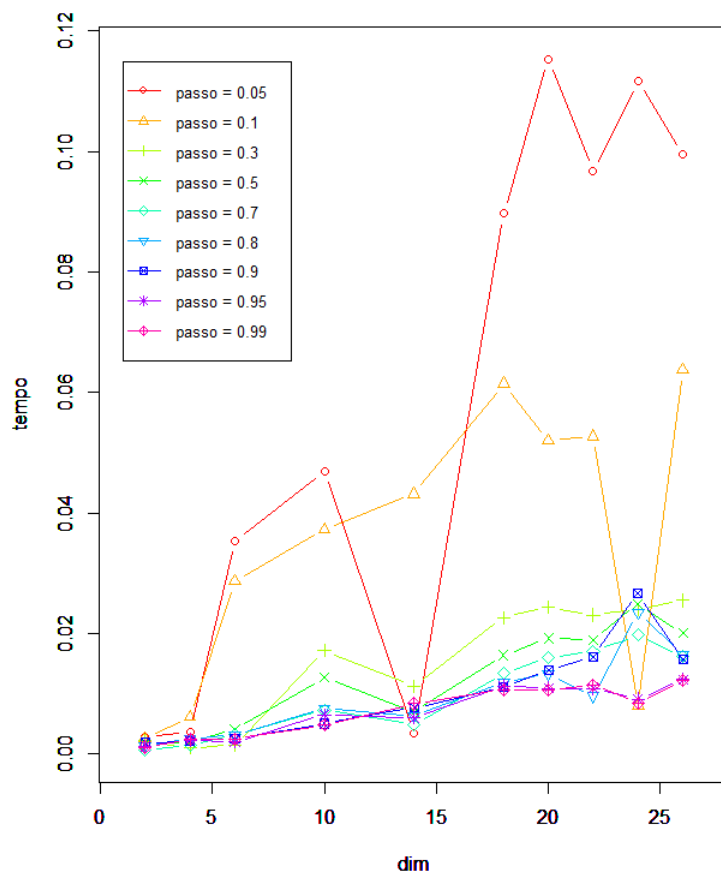
2.1.1 Tempo de execução utilizando os comandos tic e toc do Matlab (s) — PI

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	0.003453	0.004352	0.039234	0.052934	0.064244	0.108887	0.148745	0.141918	0.176101	0.186249
0,05	0.002766	0.003683	0.035443	0.046980	0.003519	0.089717	0.115260	0.096691	0.111685	0.099517
0,10	0.002524	0.006143	0.028690	0.037266	0.043178	0.061455	0.052048	0.052716	0.008070	0.063804
0,30	0.002246	0.000994	0.001510	0.017166	0.011241	0.022664	0.024393	0.023086	0.023977	0.025499
0,50	0.001641	0.002003	0.004164	0.012671	0.006954	0.016427	0.019213	0.018959	0.024918	0.020021
0,70	0.000732	0.001425	0.003236	0.007296	0.004883	0.013452	0.015941	0.017053	0.019754	0.016267
0,80	0.001296	0.002466	0.003145	0.007566	0.006263	0.011937	0.013257	0.009638	0.023427	0.016503
0,90	0.001842	0.002083	0.002493	0.005041	0.007699	0.011079	0.013924	0.016146	0.026666	0.015658
0,95	0.001275	0.002286	0.001986	0.006522	0.006000	0.011323	0.010926	0.010849	0.009118	0.012394
0,99	0.001146	0.002446	0.002477	0.004815	0.008347	0.010643	0.010549	0.011507	0.008397	0.012288

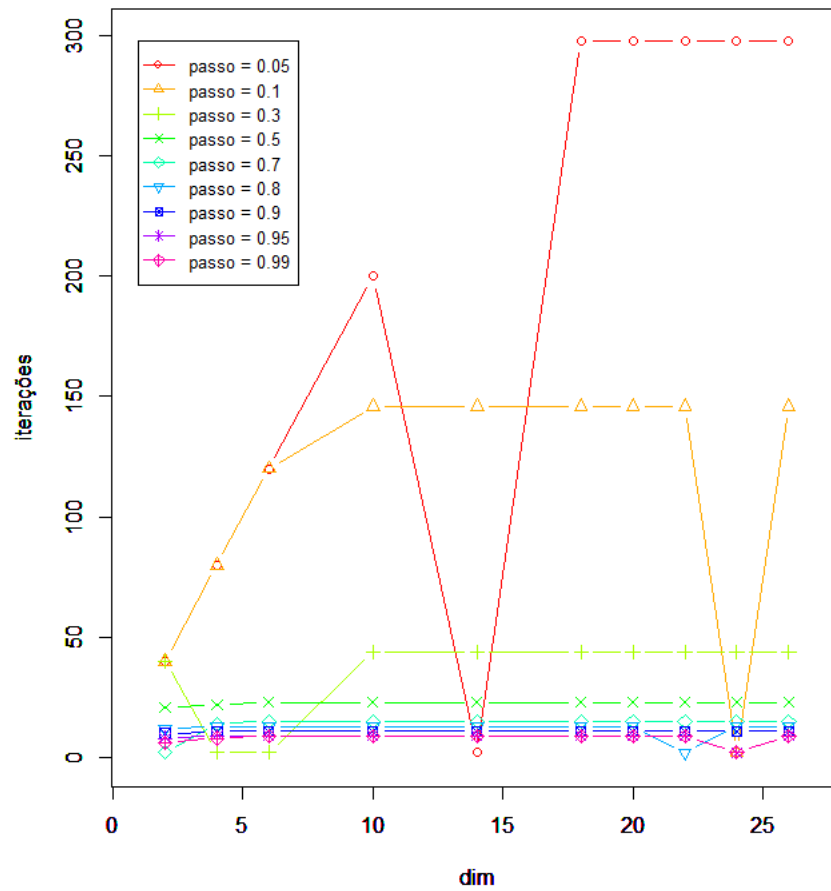
2.1.2 Número de iterações do algoritmo — PI

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	40	80	120	200	280	360	400	440	480	520
0,05	40	80	120	200	2	298	298	298	298	298
0,10	40	80	120	146	146	146	146	146	2	146
0,30	40	2	2	44	44	44	44	44	44	44
0,50	21	22	23	23	23	23	23	23	23	23
0,70	2	14	15	15	15	15	15	15	15	15
0,80	12	13	13	13	13	13	13	2	13	13
0,90	10	11	11	11	11	11	11	11	11	11
0,95	8	9	9	9	9	9	9	9	2	9
0,99	6	8	9	9	9	9	9	9	2	9

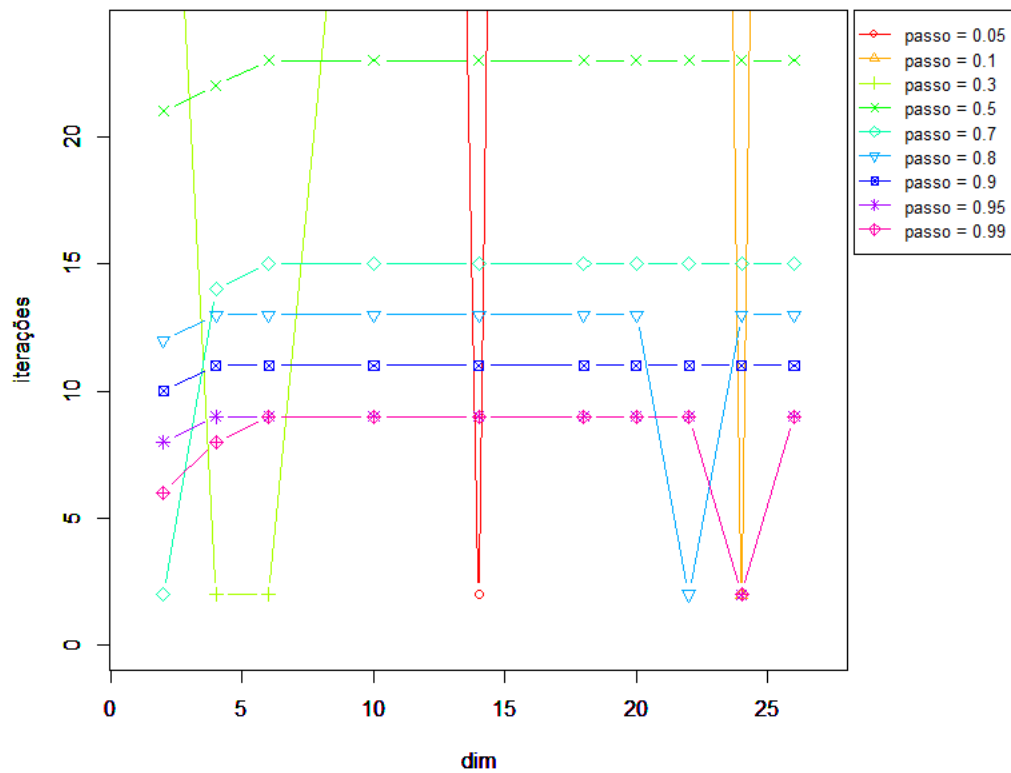
2.1.3 Gráficos Dimensão \times Tempo — PI



2.1.4 Gráficos Dimensão \times Iteração — PI



Abaixo com zoom nos números próximos de zero:



2.1.5 Erro da função objetivo (%) — PI

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	66.936766	44.793342	29.979051	13.438920	6.036714	2.723806	1.835281	1.240662	0.842560	0.575787
0,05	13.046164	1.836318	0.323758	0.009671	25	0.000099	0.000099	0.000099	0.000099	0.000099
0,10	1.795713	0.052556	0.001204	0.000095	0.000095	0.000095	0.000095	0.000095	25	0.000095
0,30	0.000127	24.8	24.992000	0.000078	0.000078	0.000078	0.000078	0.000078	0.000078	0.000078
0,50	0.000095	0.000095	0.000064	0.000064	0.000064	0.000064	0.000064	0.000064	0.000064	0.000064
0,70	20	0.000085	0.000043	0.000043	0.000043	0.000043	0.000043	0.000043	0.000043	0.000043
0,80	0.000059	0.000037	0.000053	0.000053	0.000053	0.000053	0.000053	25	0.000053	0.000053
0,90	0.000058	0.000043	0.000054	0.000055	0.000055	0.000055	0.000055	0.000055	0.000055	0.000055
0,95	0.000077	0.000050	0.000078	0.000078	0.000078	0.000078	0.000078	0.000078	25	0.000078
0,99	0.000050	0.000067	0.000031	0.000031	0.000031	0.000031	0.000031	0.000031	25	0.000031

2.1.6 Erro da solução (%) — PI

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	66.976360	44.834367	30.020069	13.479887	6.077558	2.764375	1.875606	1.280625	0.881992	0.614445
0,05	13.241472	2.023039	0.438374	0.015942	10.050378	0.000175	0.000175	0.000175	0.000175	0.000175
0,10	2.119296	0.083955	0.002094	0.000170	0.000170	0.000170	0.000170	0.000170	10.050378	10.050378
0,30	0.000194	10.050373	10.050378	0.000140	0.000140	0.000140	0.000140	0.000140	0.000140	0.000140
0,50	0.000145	0.000168	0.000115	0.000116	0.000116	0.000116	0.000116	0.000116	0.000116	0.000116
0,70	10	0.000149	0.000078	0.000079	0.000079	0.000079	0.000079	0.000079	0.000079	0.000079
0,80	0.000083	0.000067	0.000099	0.000099	0.000099	0.000099	0.000099	10.050378	0.000099	0.000099
0,90	0.000103	0.000073	0.000096	0.000096	0.000096	0.000096	0.000096	0.000096	0.000096	0.000096
0,95	0.000141	0.000083	0.000137	0.000137	0.000137	0.000137	0.000137	0.000137	10.050378	0.000137
0,99	0.000054	0.000099	0.000061	0.000062	0.000062	0.000062	0.000062	0.000062	10.050378	0.000062

2.2 Híbrido

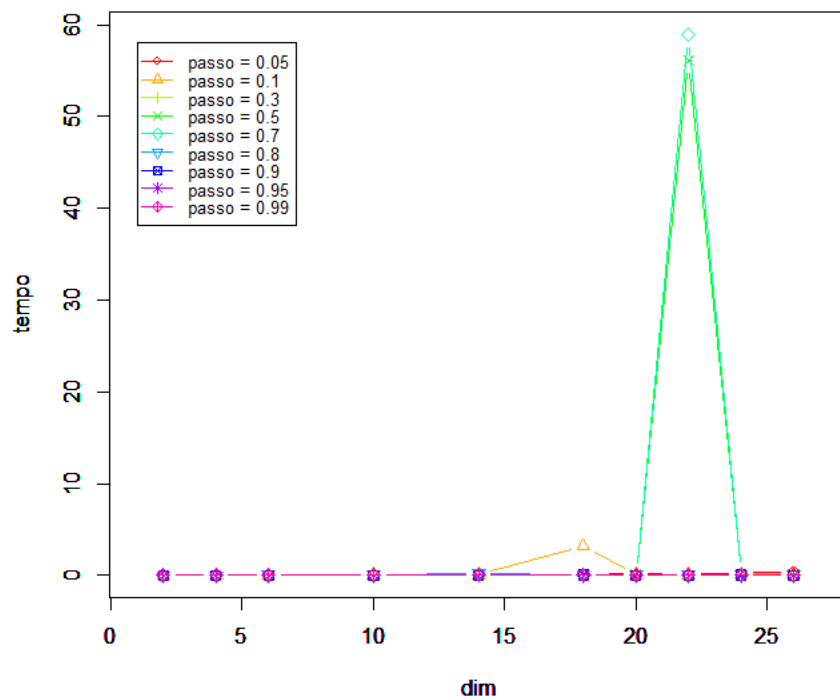
2.2.1 Tempo de execução utilizando os comandos tic e toc do Matlab (s) — Híbrido

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	0.089139	0.052131	0.088115	0.082943	0.120525	0.188099	0.149754	56.633522	0.215939	0.084534
0,05	0.007234	0.011776	0.043013	0.083892	0.112436	0.107260	0.150291	0.110571	0.217347	0.443018
0,10	0.004858	0.021237	0.035523	0.058989	0.069325	3.196353	0.069949	0.075577	0.130843	0.130140
0,30	0.003556	0.008979	0.008642	0.037804	0.032518	0.029464	0.030664	0.030757	0.057143	0.051181
0,50	0.003634	0.004616	0.026581	0.025132	0.025214	0.025949	0.037213	56.041520	0.047193	0.041327
0,70	0.002385	0.003750	0.006625	0.012313	0.030830	0.026015	0.034563	58.909649	0.045135	0.036721
0,80	0.002212	0.006508	0.009713	0.017022	0.269747	0.027347	0.033376	0.018691	0.038375	0.048149
0,90	0.002426	0.003869	0.005150	0.014072	0.018077	0.043192	0.020703	0.023014	0.061917	0.028446
0,95	0.002108	0.003645	0.006726	0.013680	0.016831	0.035256	0.018044	0.019450	0.083429	0.020458
0,99	0.003204	0.006125	0.004044	0.018070	0.017469	0.023048	0.021125	0.021285	0.043897	0.019017

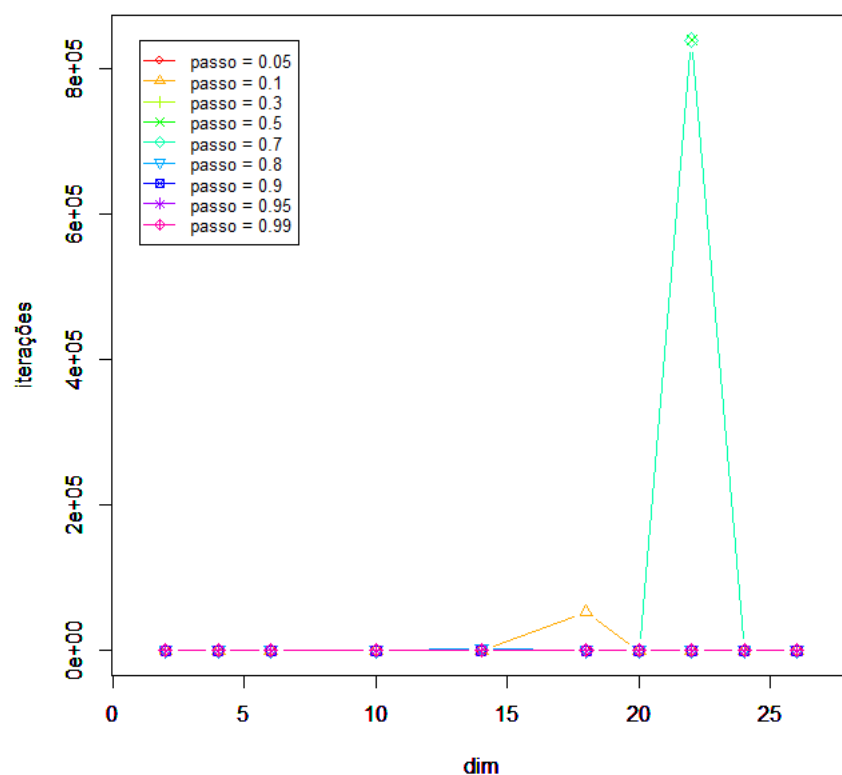
2.2.2 Número de iterações do algoritmo — Híbrido

Passo	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
0,01	42	81	121	201	281	361	401	838863	481	20
0,05	41	81	121	201	281	299	299	299	299	299
0,10	41	6	121	147	147	52431	147	147	147	147
0,30	41	44	15	45	45	45	45	45	45	45
0,50	22	23	24	24	24	24	24	838863	24	24
0,70	15	15	16	16	16	16	16	838863	16	16
0,80	13	14	14	14	3279	14	14	14	14	14
0,90	11	12	15	12	12	12	12	12	12	12
0,95	9	10	10	10	10	10	10	10	10	10
0,99	7	6	10	10	10	10	10	10	10	10

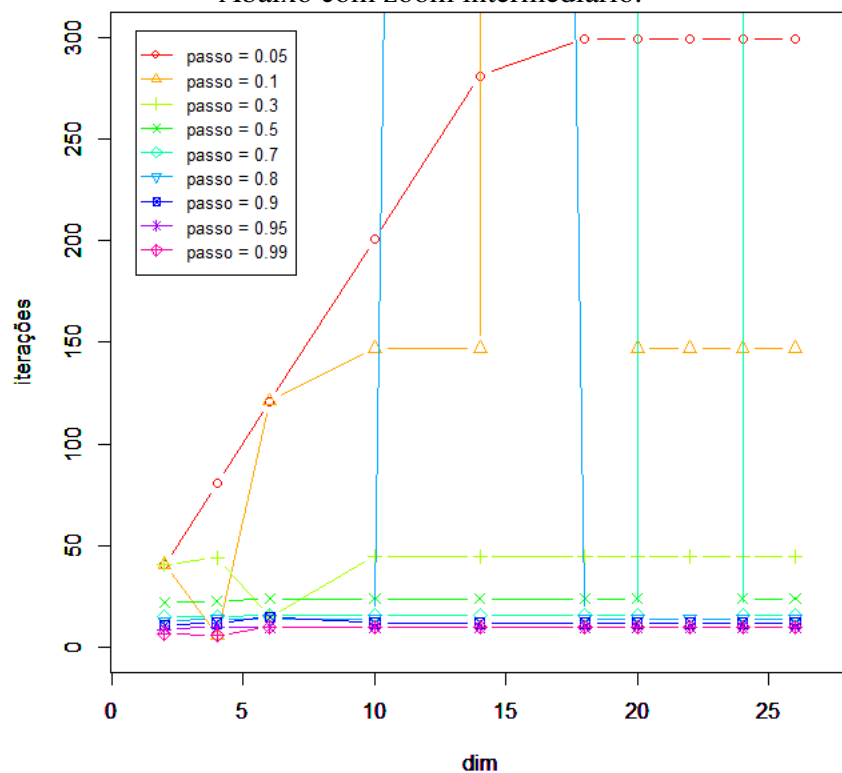
2.2.3 Gráficos Dimensão \times Tempo — Híbrido



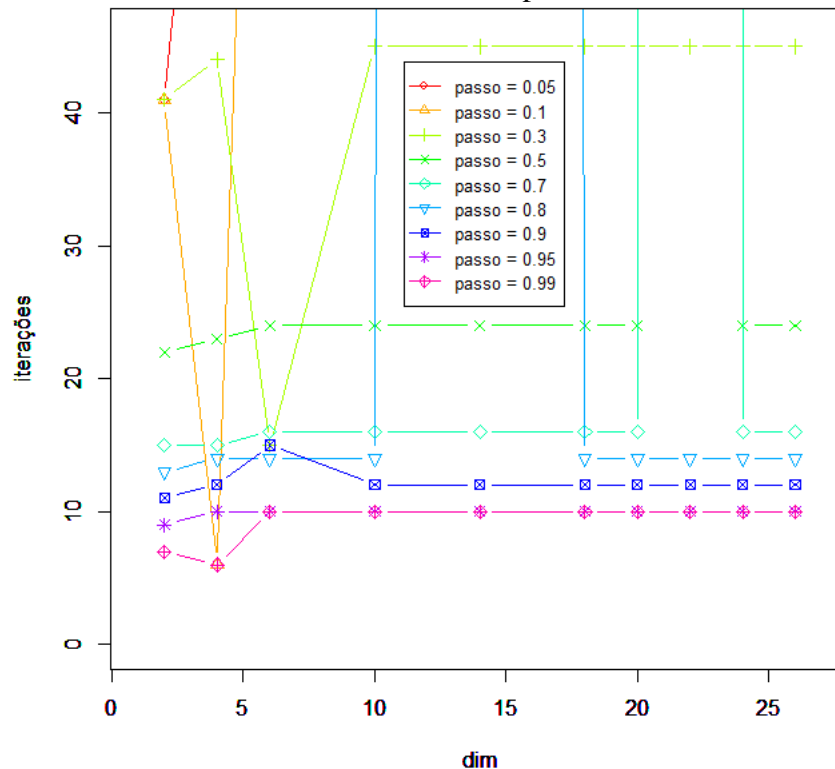
2.2.4 Gráficos Dimensão \times Iteração — Híbrido



Abaixo com zoom intermediário:



Abaixo com zoom nos números próximos de zero:



2.3 Conclusão

— Pontos Interiores:

- Tempo de execução máximo de 18,6429 centésimos de segundo. O tempo tende a aumentar com a dimensão, claro. O tempo tende a diminuir com o passo.
- Inicialmente, para passos pequenos, o número de iterações tende a aumentar com a dimensão. Contudo, esse número tende a estabilizar, ou seja, permanecer constante entre as dimensões, para passos maiores ou iguais a 0,50. Em alguns casos fortuitos, o problema é resolvido em 2 iterações. À medida que aumentamos o passo, o número de iterações vai caindo até 9.
- O erro da função objetivo não foi bom para o passo igual a 0,01. A partir daí, raramente a gente percebe um 25% patológico, mas em geral o erro está na quinta casa decimal, menor que $10^{-4}\%$.
- O erro da solução também não foi bom para o passo igual a 0,01. A partir daí, raramente a gente percebe um erro patológico na casa dos 13 ou 10%, porém em geral o erro está na quarta casa decimal, menor que 0,0002%. O erro vai diminuindo com o passo até chegar a 0,000062%.

— **Híbrido:**

- Tempo de execução máximo de 44,3018 centésimos de segundo. No entanto, houve três discrepâncias de 56,6; 56,0 e 58,9 segundos para a dimensão $n = 22$, devidas ao número de iterações do SIMPLEX.
- O número de iterações tende a estabilizar, ou seja, permanecer constante entre as dimensões, para passos maiores ou iguais a 0,05. À medida que aumentamos o passo, o número de iterações vai caindo até 10.
- Isso era de se esperar, uma vez que o Híbrido faz tudo o que o método de pontos interiores faz, e prossegue até atribuir zero ao erro, ou próximo a zero.

3. Avaliar o esforço computacional quanto a variação do gap de dualidade para os algoritmos PI e Híbrido

3.1 Pontos Interiores

3.1.1 Tempo de execução utilizando os comandos tic e toc do Matlab (s) — PI

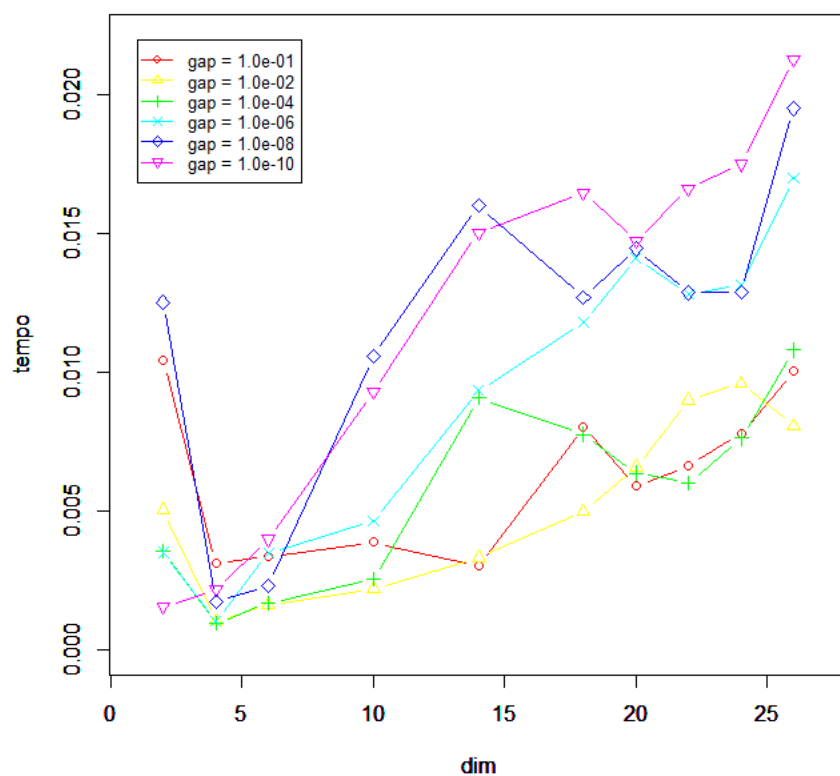
Gap*	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	0.010430	0.003138	0.003364	0.003885	0.003050	0.008023	0.005915	0.006636	0.007783	0.010038
10^{-2}	0.005056	0.001053	0.001617	0.002213	0.003334	0.005001	0.006593	0.009002	0.009619	0.008063
10^{-4}	0.003554	0.000973	0.001696	0.002548	0.009078	0.007775	0.006380	0.006018	0.007632	0.010804
10^{-6}	0.003526	0.001065	0.003508	0.004653	0.009346	0.011795	0.014112	0.012833	0.013159	0.017016
10^{-8}	0.012530	0.001756	0.002333	0.010581	0.016005	0.012718	0.014460	0.012870	0.012880	0.019503
10^{-10}	0.001559	0.002182	0.003995	0.009296	0.015038	0.016467	0.014740	0.016623	0.017508	0.021274

3.1.2 Número de iterações do algoritmo — PI

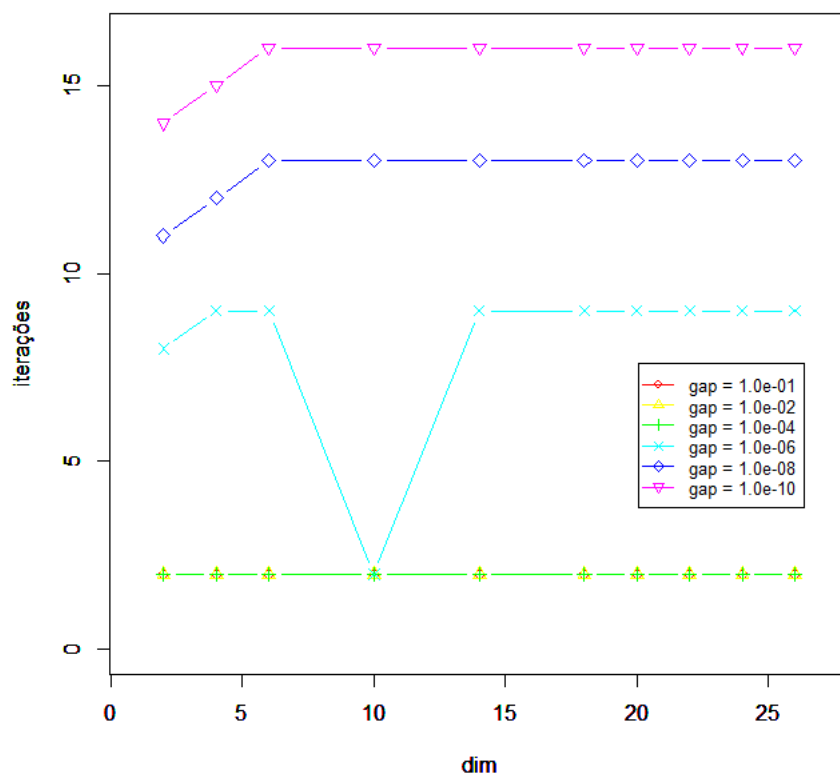
Gap	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	2	2	2	2	2	2	2	2	2	2
10^{-2}	2	2	2	2	2	2	2	2	2	2
10^{-4}	2	2	2	2	2	2	2	2	2	2
10^{-6}	8	9	9	2	9	9	9	9	9	9
10^{-8}	11	12	13	13	13	13	13	13	13	13
10^{-10}	14	15	16	16	16	16	16	16	16	16

*Gap de Dualidade

3.1.3 Gráficos Dimensão \times Tempo — PI



3.1.4 Gráficos Dimensão \times Iteração — PI



3.1.5 Erro da função objetivo (%) — PI

Gap	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	20	24.8	24.992	24.999987	25	25	25	25	25	25
10^{-2}	20	24.8	24.992	24.999987	25	25	25	25	25	25
10^{-4}	20	24.8	24.992	24.999987	25	25	25	25	25	25
10^{-6}	0.000077	0.000050	0.000078	24.999987	0.000078	0.000078	0.000078	0.000078	0.000078	0.000078
10^{-8}	0.000001	0.000001	0	0	0	0	0	0	0	0
10^{-10}	0	0	0	0	0	0	0	0	0	0

3.1.6 Erro da solução (%) — PI

Gap	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	10	10.050373	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378
10^{-2}	10	10.050373	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378
10^{-4}	10	10.050373	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378	10.050378
10^{-6}	0.000141	0.000083	0.000137	10.050378	0.000137	0.000137	0.000137	0.000137	0.000137	0.000137
10^{-8}	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001
10^{-10}	0	0	0	0	0	0	0	0	0	0

3.2 Híbrido

3.2.1 Tempo de execução utilizando os comandos tic e toc do Matlab (s) — Híbrido

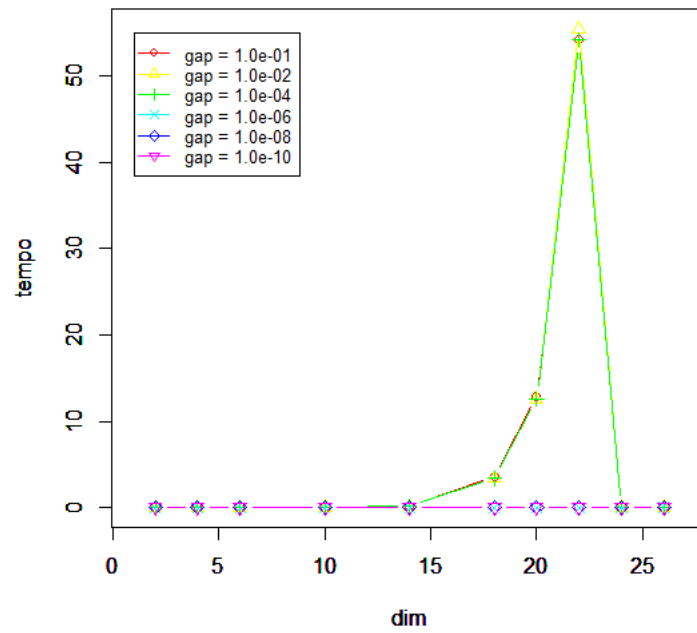
Gap [†]	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	0.001971	0.002423	0.005225	0.017462	0.182346	3.568473	12.898302	54.179670	0.019465	0.026113
10^{-2}	0.002845	0.004429	0.006313	0.035940	0.165511	3.383998	12.585765	55.448547	0.024200	0.022078
10^{-4}	0.002097	0.002966	0.004916	0.024737	0.153667	3.368345	12.594598	54.153704	0.017593	0.020116
10^{-6}	0.002182	0.002222	0.003505	0.010563	0.012200	0.013342	0.013991	0.011790	0.019556	0.022219
10^{-8}	0.002703	0.004441	0.003517	0.016393	0.010275	0.015225	0.020255	0.020362	0.026	0.024002
10^{-10}	0.003017	0.005216	0.004770	0.014446	0.013367	0.021589	0.023294	0.020789	0.025813	0.023459

3.2.2 Número de iterações do algoritmo — Híbrido

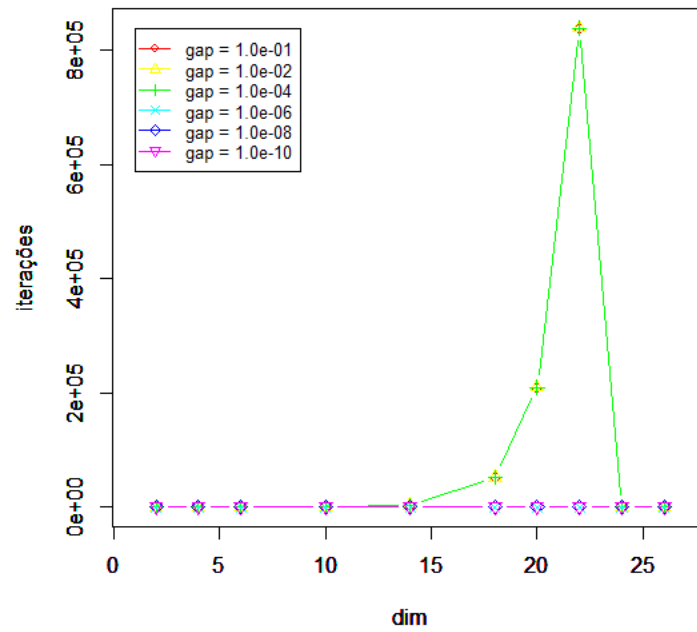
Gap	$n = 2$	$n = 4$	$n = 6$	$n = 10$	$n = 14$	$n = 18$	$n = 20$	$n = 22$	$n = 24$	$n = 26$
10^{-1}	3	6	15	207	3279	52431	209718	838863	20	14
10^{-2}	3	6	15	207	3279	52431	209718	838863	20	14
10^{-4}	3	6	15	207	3279	52431	209718	838863	20	14
10^{-6}	9	10	10	10	10	10	10	10	10	10
10^{-8}	12	13	14	14	14	14	14	14	14	14
10^{-10}	15	16	17	17	17	17	17	17	17	17

*Gap de Dualidade

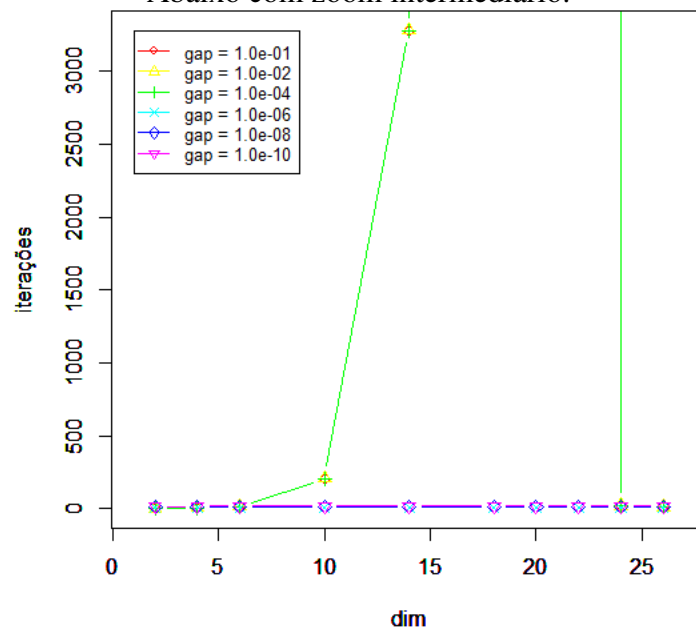
3.2.3 Gráficos Dimensão \times Tempo — Híbrido



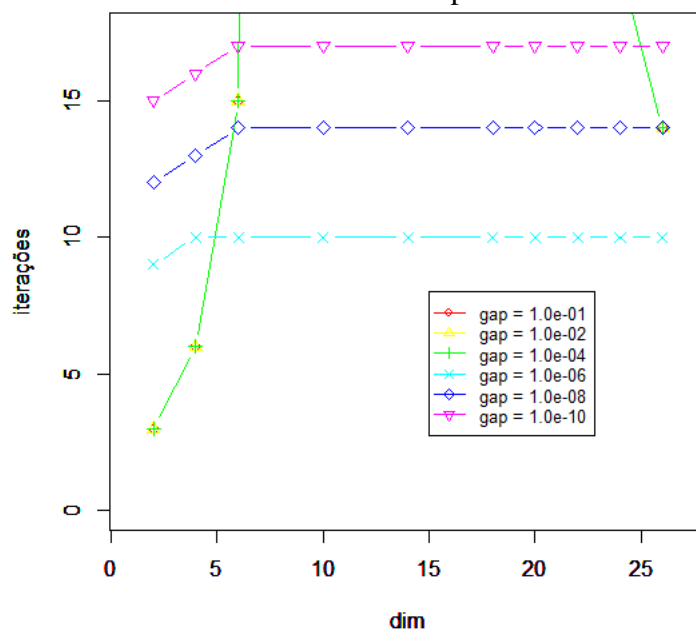
3.2.4 Gráficos Dimensão \times Iteração — Híbrido



Abaixo com zoom intermediário:



Abaixo com zoom nos números próximos de zero:



3.3 Conclusão

— Pontos Interiores:

- Tempo de execução máximo de 21,274 centésimos de segundo. O tempo tende a aumentar com a dimensão, claro. Na dimensão $n = 26$, o tempo diminui com o passo na seguinte ordem de gap de dualidade: 10^{-10} , 10^{-8} , 10^{-6} , 10^{-4} , 10^{-1} , e em 10^{-2} *atinge omnino*.
- O número de iterações tende a estabilizar, ou seja, permanecer constante entre as dimensões, para todos os gaps. Em um único caso fortuito, o problema foi resolvido em 2 iterações. À medida que diminuimos o gap, o número de iterações tende a aumentar.
- O erro da função objetivo não foi bom para gaps maiores ou iguais a 10^{-4} . A partir daí, raramente a gente percebe um 25% patológico, mas em geral o erro está na quinta casa decimal, menor que 0,00008%. À medida que baixamos o gap, o erro diminui até zerar.
- O erro da solução também não foi bom para gaps maiores ou iguais a 10^{-4} . A partir daí, raramente a gente percebe um erro patológico na casa dos 10%, porém em geral o erro está na quarta casa decimal, menor que 0,0002%. À medida que baixamos o gap, o erro diminui até zerar.

— Híbrido:

- Tempo de execução máximo de 55,5 segundos, devido às iterações do SIMPLEX. Entretanto, descontadas as raras discrepâncias maiores que 3 segundos, esse tempo máximo cai para 26,113 centésimos de segundo.
- O número de iterações tende a estabilizar, ou seja, permanecer constante entre as dimensões, para gaps menores ou iguais a 10^{-6} . À medida que diminuimos o gap, o número de iterações aumenta de 10 a 17, sendo que obtivemos mínimo para o gap de 10^{-6} .

4. Resultados relacionados ao esforço computacional para o algoritmo Híbrido (Detalhamento dos métodos de Pontos Interiores e SIMPLEX)

Dimensão	Iterações		Tempo (s)		Erro f_{obj} (%)		Erro Solução (%)	
	PI	SIMPLEX	PI	SIMPLEX	PI	SIMPLEX	PI	SIMPLEX
2	2	1	0.039462	0.001268	20	0	10	0
4	9	1	0.002359	0.001790	0.000050	0	0.000083	0
6	9	1	0.002195	0.006854	0.000078	0	0.000137	0
10	9	1	0.006234	0.005267	0.000078	0	0.000137	0
14	9	1	0.020896	0.005505	0.000078	0	0.000137	0
18	9	1	0.010790	0.005870	0.000078	0	0.000137	0
20	9	1	0.009148	0.004800	0.000078	0	0.000137	0
22	9	1	0.010310	0.009628	0.000078	0	0.000137	0
24	9	1	0.008463	0.006871	0.000078	0	0.000137	0
26	9	1	0.014276	0.011024	0.000078	0	0.000137	0

A grande questão relacionada ao algoritmo Híbrido é quando se deve chavear do algoritmo PI para o SIMPLEX.

Consideramos gap de dualidade igual a 10^{-6} e passo igual a 0,95. Foi em ambos esses casos que obtivemos melhor número de iterações.

Ao detalharmos, verificamos que o PI executa sempre em 9 iterações ou menos. O tempo de execução obtido foi sempre 3,9462 centésimos de segundo ou menos. O erro da função objetivo estabilizou em 0,000078%. E o erro da solução estabilizou em 0,000137%.

A partir daí, roda o SIMPLEX sempre em única iteração. O tempo adicional foi sempre de 0,9628 centésimos de segundo ou menos. Os erros da função objetivo e da solução caíram para zero.

5. Ponto Inicial e Parâmetros do Problema de Klee-Minty

O ponto inicial e a constante valb do problema de Klee-Minty não são considerados no SIMPLEX. São utilizados no PI; e, por consequência, também no Híbrido.

Consideramos o código em MatLab:

```
valb = 10;
for il = 1:n
    b(il,1) = valb^(il-1);
    xmax(il,1) = b(il,1);
end
Xini = (1.0e-11 + 1.0e-08*rand(1)) * (XMAX./norm(XMAX));
```

O ponto inicial considerado é próximo da origem. É um unitário multiplicado por um aleatório perturbado. Não poderia ser diferente, uma vez que o problema segue superfícies de nível e deve começar das menores para as maiores. O que poderia ser feito é determinar uma superfície de nível mínima conhecida para acelerar o processo.

O vetor valb considerado é $(1, 10, 100, 1000, \dots, 10^n)^\top$.

Rodei a última parte do detalhamento do Híbrido para valb igual a 100, 1000 e 10000 e os números de iterações se mantiveram em 9 + 1 em geral. Contudo, houve uma única discrepância, exibida abaixo:

```
-----
valb                                = 1000
Dimensão                            = 20
Gap de dualidade                     = 1.000000e-06
Passo                                = 0.950000
PI erro percentual optimal objective value is = 25.000000
PI erro percentual vetor x solução    = 10.050378
Elapsed time is 0.007506 seconds.
Theoretical number of iterations for the SIMPLEX is = 1048576
PI number of iterations is            = 2
SIMPLEX number of iterations is       = 209716
SIMPLEX erro percentual optimal objective value is = 0.000000
SIMPLEX erro percentual vetor x solução = 0.000000
Elapsed time is 12.627736 seconds.
-----
```

Isso significa que execuções consecutivas são capazes de eliminar as discrepâncias. O que se pode fazer é interromper o algoritmo após milhares de iterações e reiniciar o algoritmo.

Também concluímos que a escolha do gap de dualidade e do passo é suficiente para resolver o problema, para qualquer valor tolerável de valb, de forma idêntica ao caso considerado.

6. Conclusão

Gostaríamos de deixar como observação que: o zero “computacional” que aparece neste trabalho é um número de módulo menor que 0,00000049, arredondado para baixo.

Como foi dito, o SIMPLEX é exatamente preciso, fica incomparavelmente lento ao aumentar a dimensão do problema, pois converge em muitas iterações.

Resta-nos comparar o método de Pontos Interiores com o Híbrido. Após variarmos o passo e o gap de dualidade, selecionamos os de melhor desempenho e concluímos que:

O PI executado separadamente já é preciso. Consegue boa taxa de convergência em 9 iterações. Acompanhado do SIMPLEX, O Híbrido é exato.

O PI rodado à parte é rápido. Acompanhado do SIMPLEX, o Híbrido em geral continua rápido; mas deve ser policiado, porque de vez em quando pode ficar lento. Raramente há impacto com o aumento da dimensão do problema.

Versão de 08/dezembro/2021[‡] por Vinicius Claudino Ferraz.

Matrícula = 2019435823.

[‡]Fora da caridade não há salvação.