Vector Transport Free Riemannian LBFGS for Optimization on Symmetric Positive Definite Matrix Manifolds (Supplementary Material)

Reza Godaz* REZA.GODAZ@MAIL.UM.AC.IR
Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

Benyamin Ghojogh* BGHOJOGH@UWATERLOO.CA
Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada

Reshad Hosseini RESHAD.HOSSEINI@UT.AC.IR
Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran

Reza Monsefi MONSEFI@UM.AC.IR
Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

Fakhri Karray KARRAY@UWATERLOO.CA
Mark Crowley MCROWLEY@UWATERLOO.CA
Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada

Editors: Vineeth N Balasubramanian and Ivor Tsang

1. Proof for Proposition 5

– Upper bound analysis: The most time consuming part of the RLBFGS algorithm is the recursive function GetDirection(.) defined in Section 2.4. In every call of recursion, Eqs. (1)-(3) are performed. The metric used in Eqs. (1) and (3) takes $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$ before and after applying the mappings, respectively (cf. Table 1). This is because matrix inversion takes $\mathcal{O}(n^3)$ time and the matrices are vectorized within trace operator in the algorithm implementation. The vector transport used in Eq. (2) takes $\mathcal{O}(n^3)$ and $\mathcal{O}(1)$ before and after applying the mappings, respectively (cf. Table 1). Recursion in RLBGS is usually done for a constant m number of times (see Ref. Ring and Wirth, 2020). Overall, the complexity of recursion is $\mathcal{O}(mn^3)$ and $\mathcal{O}(mn^2)$, for RLBFGS with and without mappings, respectively.

– Lower bound analysis: Computing matrix inversion in the metric before applying our mappings (cf. Table 1) takes $\Omega(n^2 \log n)$ (see article [1] referenced below) while metric takes $\Omega(n^2)$ after the mappings. The vector transport takes $\Omega(n^3)$ and $\Omega(1)$ before and after the mappings, respectively. Overall, the complexity of recursion is $\Omega(mn^3)$ and $\Omega(mn^2)$, for RLBFGS with and without mappings, respectively.
[1] A. Tveit, "On the complexity of matrix inversion," Norwegian University of Science and Technology, Trondheim, Technical Report, 2003.

* The first two authors contributed equally to this work.

– **Tight bound analysis:** As the upper bound and lower bound complexities of the every algorithm are equal, we can conclude that the complexity bound is tight. Therefore, RLBFGS has time complexity $\Theta(mn^3)$ and $\Theta(mn^2)$, with and without our mappings, respectively. This shows that our proposed mappings improve the time of optimization. This time improvement shows off better if computation of gradients in Eq. (6) is not dominant in complexity.

2. Simulations for Larger Sample Size and Dimensionality

Tables 2 and 3, in this Supplementary Material, report the average simulation results for large sample size, i.e., $N = 100n^2$. In these tables, exponential map and Taylor approximation for retraction are used, respectively. We also have reported results for large dimensionality $d = 100$ in these two tables.

3. Cost Difference Progress for $N = 10n^2$ Sample Size Simulations

The log-scale cost difference for simulations of $N = 10n^2$ are depicted in Figs. 1 and 2 of this Supplementary Material, for $K = 2$ and $K = 5$, respectively.

4. Cost Difference Progress for $N = 100n^2$ Sample Size Simulations

The log-scale cost difference for simulations of $N = 10n^2$ are depicted in Figs. 3 and 4 of this Supplementary Material, where exponential map and Taylor approximation for retraction are used, respectively.

Table 1: Comparison of average results over ten runs where retraction with Taylor series expansion is used in algorithms and $K \in \{2,5\}$, $n \in \{2,10\}$, $N = 10n^2 \in \{40, 1000\}$. The #iters, conv, iter, diff, and std are short for number of iterations, convergence, iteration, difference, and standard deviation, respectively.

K	n	Separation	Algorithm	#iters	conv. time	time diff. std	iter. time	last cost
2	2	Low	VTF (ISR)	59.800±22.553	93.135±88.437	19.280	1.320±0.711	0.610±0.366
			VTF (Chol.)	62.600±29.079	106.545±117.633	48.611	1.355±0.830	0.619±0.382
			RLBFGS	59.800±26.803	100.424±120.310	–	1.363±0.787	0.610±0.366
		Mid	VTF (ISR)	45.800±16.982	48.155±38.045	10.833	0.900±0.454	0.482±0.497
			VTF (Chol.)	47.000±18.074	51.529±41.465	11.146	0.930±0.483	0.482±0.497
			RLBFGS	45.000±16.330	48.150±37.285	–	0.921±0.458	0.482±0.497
		High	VTF (ISR)	25.600±4.142	9.816±3.837	1.380	0.370±0.093	0.270±0.456
			VTF (Chol.)	26.300±4.448	10.540±4.190	1.932	0.385±0.101	0.270±0.456
			RLBFGS	25.500±4.353	10.526±4.819	–	0.396±0.113	0.270±0.456
	10	Low	VTF (ISR)	75.800±25.607	166.736±137.600	63.108	1.955±0.816	4.129±0.771
			VTF (Chol.)	74.500±22.629	152.074±107.531	98.424	1.855±0.682	4.129±0.771
			RLBFGS	74.800±25.442	172.641±143.198	–	2.054±0.834	4.129±0.771
		Mid	VTF (ISR)	50.900±13.956	64.501±38.055	11.794	1.170±0.395	3.472±1.154
			VTF (Chol.)	52.100±13.892	66.775±41.257	7.815	1.184±0.410	3.472±1.154
			RLBFGS	51.000±14.063	70.132±46.421	–	1.264±0.450	3.472±1.154
		High	VTF (ISR)	43.600±5.522	42.763±11.897	12.828	0.963±0.168	4.353±1.081
			VTF (Chol.)	47.400±6.620	50.438±15.187	8.731	1.041±0.182	4.353±1.081
			RLBFGS	44.300±6.464	47.830±14.747	–	1.055±0.192	4.353±1.081
5	2	Low	VTF (ISR)	262.500±126.532	4430.577±4455.877	13196.851	13.849±6.991	0.082±0.281
			VTF (Chol.)	253.500±124.970	4086.076±3967.586	10698.273	13.084±6.850	0.099±0.279
			RLBFGS	270.700±144.145	5305.821±5495.383	–	15.560±8.452	0.090±0.282
		Mid	VTF (ISR)	110.900±50.886	731.318±769.639	184.154	5.445±2.806	1.010±0.349
			VTF (Chol.)	112.200±60.736	792.756±1053.832	416.142	5.436±3.358	1.010±0.349
			RLBFGS	111.900±55.183	814.994±975.737	–	5.827±3.289	1.010±0.349
		High	VTF (ISR)	52.000±13.565	116.485±69.535	12.185	2.071±0.728	1.626±0.431
			VTF (Chol.)	51.400±12.616	114.032±67.916	19.838	2.057±0.735	1.626±0.431
			RLBFGS	53.400±14.081	139.989±89.213	–	2.408±0.936	1.626±0.431
	10	Low	VTF (ISR)	219.700±57.908	2946.040±1513.372	1239.375	12.579±3.507	6.643±0.662
			VTF (Chol.)	226.100±86.010	3272.111±2808.634	2103.162	12.707±5.156	6.625±0.650
			RLBFGS	231.300±64.484	3607.215±1947.580	–	14.516±4.305	6.642±0.663
		Mid	VTF (ISR)	87.100±21.502	413.616±223.510	54.437	4.458±1.310	6.585±0.569
			VTF (Chol.)	87.200±21.186	405.484±214.985	47.791	4.374±1.262	6.585±0.569
			RLBFGS	87.400±20.403	454.434±227.556	–	4.918±1.339	6.585±0.569
		High	VTF (ISR)	60.500±10.533	181.113±76.070	16.691	2.892±0.649	6.827±0.836
			VTF (Chol.)	62.700±11.295	197.565±93.653	24.531	3.019±0.827	6.827±0.836
			RLBFGS	58.900±11.040	187.518±86.254	–	3.058±0.746	6.827±0.836

Table 2: Comparison of average results over ten runs where exponential map is used in algorithms and $n \in \{2, 10, 100\}$, $N = 100n^2 \in \{400, 10000, 1000000\}$, $K = 2$.

n	Separation	Algorithm	#iters	conv. time	time diff. std	iter. time	last cost
2	Low	VTF (ISR)	72.000±22.949	127.902±98.372	31.248	1.616±0.584	0.281±0.394
		VTF (Chol.)	69.800±23.136	116.250±86.811	23.937	1.490±0.590	0.281±0.394
		RLBFGS	68.900±24.875	123.064±105.594	–	1.561±0.694	0.281±0.394
	Mid	VTF (ISR)	59.400±16.460	78.124±53.880	10.139	1.210±0.421	0.520±0.392
		VTF (Chol.)	54.900±12.862	63.241±34.382	31.027	1.082±0.332	0.516±0.393
		RLBFGS	58.400±17.037	80.042±62.183	–	1.240±0.497	0.520±0.392
	High	VTF (ISR)	23.300±2.627	7.434±2.288	0.738	0.313±0.057	0.102±0.288
		VTF (Chol.)	22.900±2.685	7.273±2.084	0.873	0.312±0.052	0.102±0.288
		RLBFGS	23.300±2.497	7.917±2.135	–	0.335±0.053	0.102±0.288
10	Low	VTF (ISR)	63.400±16.728	112.682±74.903	17.031	1.664±0.489	4.364±1.299
		VTF (Chol.)	65.000±17.994	117.215±82.258	13.637	1.670±0.538	4.364±1.299
		RLBFGS	64.900±17.866	126.987±87.668	–	1.819±0.561	4.364±1.299
	Mid	VTF (ISR)	48.200±8.766	58.096±23.089	10.899	1.164±0.256	4.024±1.627
		VTF (Chol.)	49.800±11.811	64.021±34.770	6.571	1.208±0.364	4.024±1.627
		RLBFGS	48.400±10.906	64.099±32.746	–	1.251±0.361	4.024±1.627
	High	VTF (ISR)	45.100±7.385	49.628±20.550	7.336	1.065±0.243	3.290±1.129
		VTF (Chol.)	47.800±8.121	55.217±22.622	6.572	1.117±0.251	3.290±1.129
		RLBFGS	45.200±7.131	52.883±20.627	–	1.137±0.236	3.290±1.129
100	Low	VTF (ISR)	131.900±32.402	54826.327±30454.295	5842.849	389.761±121.060	63.703±2.843
		VTF (Chol.)	141.200±36.908	59082.278±32385.087	3775.919	392.658±110.916	63.703±2.843
		RLBFGS	136.300±36.059	56627.788±32602.224	–	387.367±119.404	63.703±2.843
	Mid	VTF (ISR)	87.800±20.225	23635.623±11743.237	5738.406	259.026±52.061	61.263±4.734
		VTF (Chol.)	95.800±25.170	28033.655±16196.321	1410.113	278.456±64.670	61.263±4.734
		RLBFGS	94.900±25.653	28527.524±17341.755	–	284.548±69.821	61.263±4.734
	High	VTF (ISR)	97.000±24.240	29048.836±12857.812	3441.882	286.602±63.250	61.241±3.991
		VTF (Chol.)	105.100±27.477	33967.543±15707.841	2014.574	308.668±70.065	61.241±3.991
		RLBFGS	103.200±26.389	33101.986±15790.066	–	305.002±74.361	61.241±3.991

Table 3: Comparison of average results over ten runs where retraction with Taylor series expansion is used in algorithms and $n \in \{2, 10, 100\}$, $N = 100n^2 \in \{400, 10000, 1000000\}$, $K = 2$.

n	Separation	Algorithm	#iters	conv. time	time diff. std	iter. time	last cost
2	Low	VTF (ISR)	79.800±49.973	206.935±343.153	105.106	1.839±1.340	0.403±0.578
		VTF (Chol.)	72.200±20.004	125.636±83.113	334.239	1.608±0.536	0.402±0.576
		RLBFGS	83.900±51.054	237.936±364.382	–	2.064±1.412	0.404±0.578
	Mid	VTF (ISR)	48.100±12.688	49.277±30.486	12.185	0.946±0.334	0.196±0.436
		VTF (Chol.)	50.100±14.271	56.329±40.420	14.674	1.017±0.422	0.196±0.436
		RLBFGS	51.000±13.622	61.036±39.550	–	1.099±0.412	0.196±0.436
	High	VTF (ISR)	25.300±3.945	9.887±3.903	3.565	0.377±0.101	0.111±0.261
		VTF (Chol.)	26.500±4.927	11.003±5.644	4.938	0.395±0.124	0.111±0.261
		RLBFGS	25.100±4.012	10.203±4.224	–	0.392±0.106	0.111±0.261
10	Low	VTF (ISR)	65.500±18.435	123.552±77.898	14.558	1.747±0.560	4.164±1.142
		VTF (Chol.)	66.500±19.558	122.281±79.423	19.994	1.688±0.569	4.164±1.142
		RLBFGS	65.200±18.743	130.098±84.241	–	1.835±0.622	4.164±1.142
	Mid	VTF (ISR)	40.600±5.211	38.915±12.591	8.862	0.938±0.177	4.589±1.278
		VTF (Chol.)	41.000±5.538	38.732±12.699	9.319	0.924±0.171	4.589±1.278
		RLBFGS	40.400±6.186	41.513±15.303	–	0.998±0.213	4.589±1.278
	High	VTF (ISR)	44.400±6.310	46.957±15.325	11.008	1.032±0.198	3.786±1.113
		VTF (Chol.)	46.000±6.616	50.393±15.950	12.688	1.070±0.195	3.786±1.113
		RLBFGS	44.200±7.099	50.707±18.070	–	1.116±0.222	3.786±1.113
100	Low	VTF (ISR)	118.500±11.404	39986.299±6719.992	2227.745	335.341±26.657	62.241±5.708
		VTF (Chol.)	124.400±11.157	42882.658±7219.289	3669.708	342.538±28.830	62.241±5.708
		RLBFGS	121.300±11.235	41171.059±7987.985	–	336.547±36.783	62.241±5.708
	Mid	VTF (ISR)	103.800±24.679	32533.037±16124.089	3314.944	297.268±76.267	60.256±3.662
		VTF (Chol.)	111.900±28.781	37328.925±18924.066	2916.610	314.512±82.724	60.256±3.662
		RLBFGS	111.300±27.941	36926.995±17705.262	–	314.564±76.642	60.256±3.662
	High	VTF (ISR)	96.300±25.880	28818.656±14923.845	1492.788	283.824±67.267	60.601±4.436
		VTF (Chol.)	103.900±26.126	33669.987±16494.757	3368.579	308.353±73.281	60.601±4.436
		RLBFGS	102.800±25.258	31630.478±15755.934	–	292.771±68.396	60.601±4.436

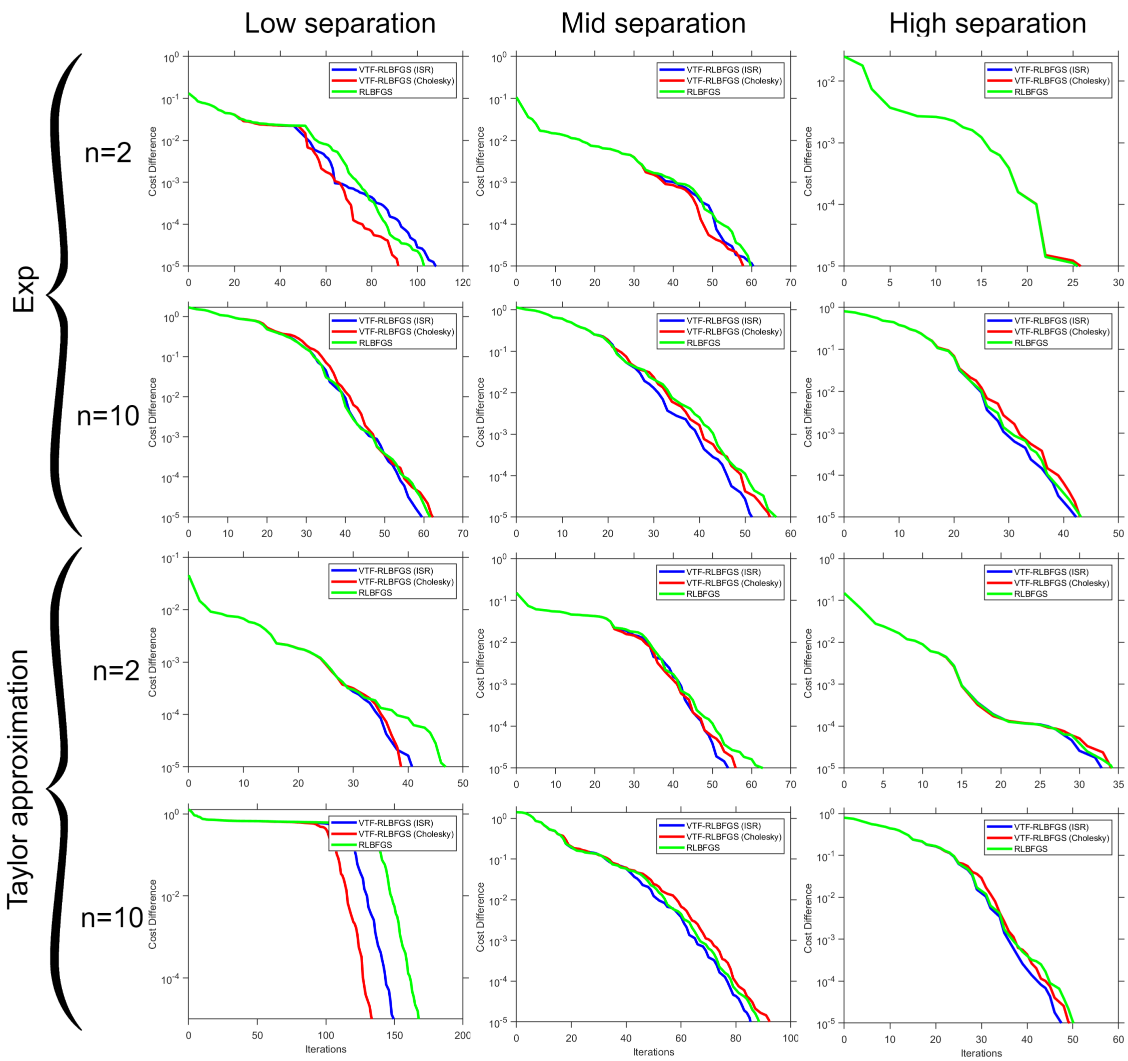

Figure 1: Comparison of the proposed VTF-RLBFGS algorithm (using ISR and Cholesky) with RLBFGS in their cost differences. In these experiments, we had $K = 2$.

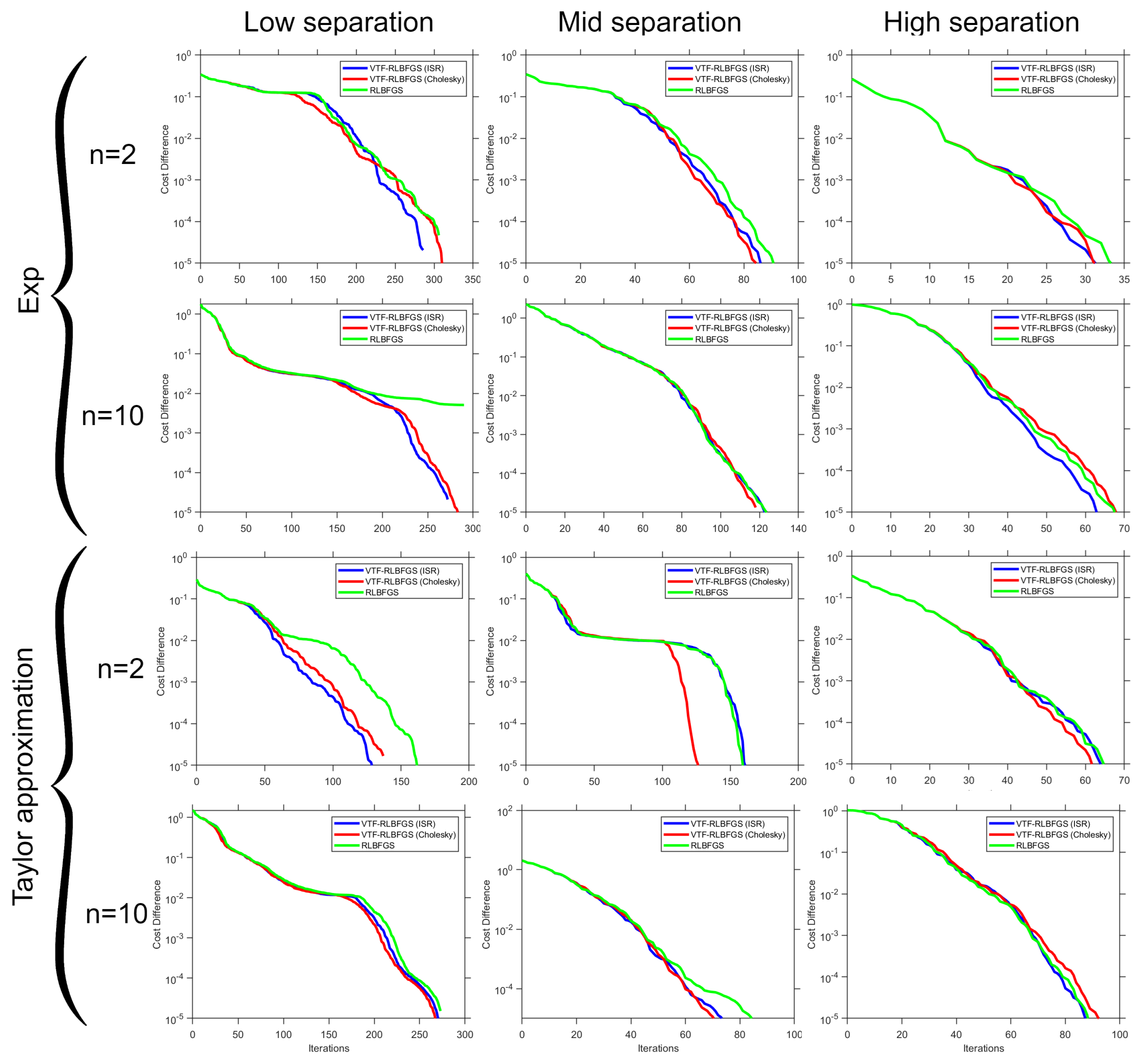

Figure 2: Comparison of the proposed VTF-RLBFGS algorithm (using ISR and Cholesky) with RLBFGS in their cost differences. In these experiments, we had $K = 5$.

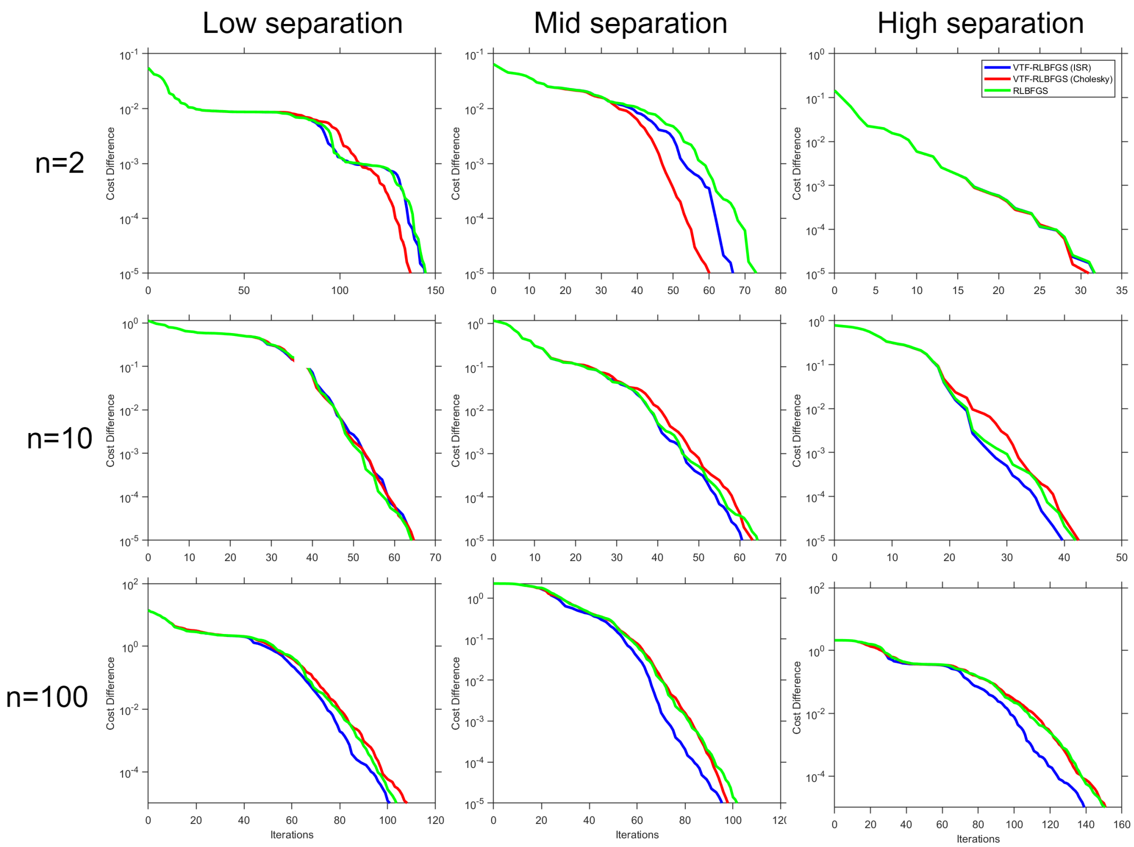

Figure 3: Comparison of the proposed VTF-RLBFGS algorithm (using ISR and Cholesky) with RLBFGS in their cost differences. In these experiments, exponential map is used in optimization procedure and we had $K = 2$.

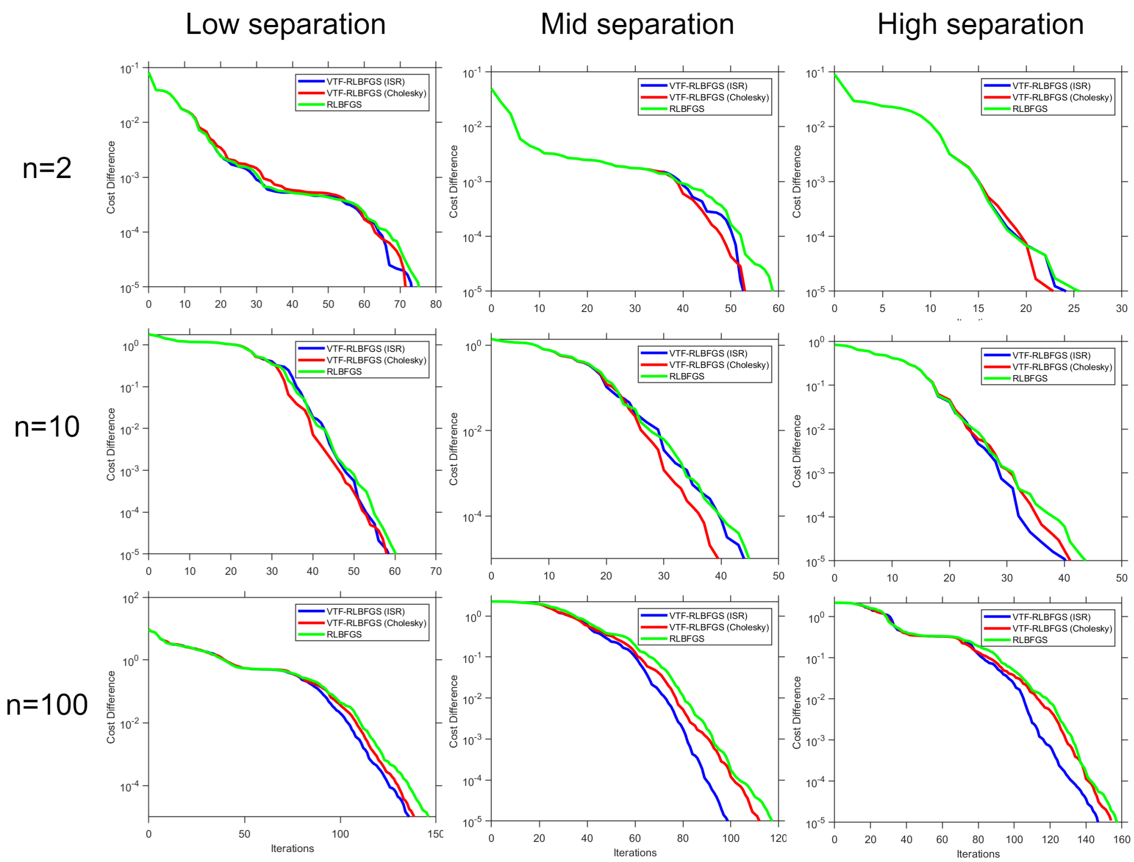

Figure 4: Comparison of the proposed VTF-RLBFGS algorithm (using ISR and Cholesky) with RLBFGS in their cost differences. In these experiments, Taylor series expansion is used for approximation of retraction operator and we had $K = 2$.