Unconstrained Optimization

Sara Rosato
Politecnico di Torino
s317547

Florentin-Cristian Udrea
Politecnico di Torino
s319029

***Abstract*—In this review we're going to examine and compare the results obtained using tree numerical methods for unconstrained optimization: Steepest Descent Method and Fletcher & Reeves which uses exact computations and Inexact Newton Method with Finite Differences which uses mostly approximated computations.**

I. Introduction

The aim of unconstrained optimization is to find x that minimizes $f(x)$, with $x \in R^n$ and f: $R^n \rightarrow R$, having no constraints on x.

Steepest descent Method (SD) is an optimization method that moves along the direction

$$p_k = -\nabla f(x_k)$$

at each step. This one requires only the computation of the gradient value and this choice is convenient as a search direction, since it is the one along which f decreases most rapidly from the point of evaluation.

The **Conjugate Gradient Method**, instead, improves the SD method by computing a descent direction that is never orthogonal to any of the previous decent directions. It can be proven that using such a descent directions leads to a faster convergence respect to the SD [2]. **Fletcher and Reeves' method** (CG-FR) adapts the conjugate gradient in order to minimize general non-linear functions. The differences are the step length which is computed through a line search that look for an approximate minimum of the function along the direction p_k and the definition of the residual that now is the gradient of f:

$$\beta^{k+1} = \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k} = \frac{||\nabla f_{k+1}||^2}{||\nabla f_k||^2}$$

$$p_{k+1} = -\nabla f_{k+1} + \beta^{k+1} p^k$$

For **Inexact Newton method** (INM), instead, p_k has to satisfy the following:

$$||\nabla^2 f(x_k)p + \nabla f(x_k)|| \leq \eta_k ||\nabla f(x_k)||$$

where η_k are "forcing terms" and, depending on which one of them is chosen, a different rate of convergence is obtained. In this way the residual is not zero but a small quantity, depending on the gradient. The method therefore uses approximations in order to reduce the computational cost, which is quite large for Newton Method. Moreover we also implemented 'Finite Differences' to our Inexact Newton Method, which is a technique that allows us to skip the full computation of the gradient and hessian matrix by approximating them.

In all the methods we've implemented a backtracking strategy with Armijo condition, using as parameters ρ = 0.5 and c = 10^{-4}. While for the INM we always started with a step α_0 = 1 (the construct), for the SD and CG-FR methods we chose an α_0 = 5. This choice is motivated by the different way the descent vectors are computed.

At first we tested our methods on the Rosenbrock function, then attempted to solve the problems number 5, 13, 14 and 16 chosen among those listed in [1] having a dimension of $n = 10^4$.

Since we used the forcing term $\eta_k = min(0.5, \sqrt{||\nabla f(x_k)||})$ we expect a superlinear convergence rate from the INM, thus comparable with CG-FR and higher than the linear convergence rate of the SD. Furthermore we imposed a convergence tolerance $tol = 10^{-6}$, meaning we consider a method to converge if $||\nabla f(x_k)|| < tol$.

In introducing the problems we used the notation $\overline{x_i}$ to indicate the starting point.

II. Rosenbrock function

$$f(x_1, x_2) = (1 - x_1)^2 + 100\left(x_2 - x_1^2\right)^2$$

$$\overline{x_{(0)}} = (1.2, 1.2), \overline{x_{(1)}} = (-1.2, 1)$$

The Rosenbrock function is a non-convex function $f : R^2 \rightarrow R$ which is often used as a test problem for optimization algorithms. It has a global minimum of 0 at the point x^*=(1, 1).
We tested our methods with two different starting points: $\overline{x_{(0)}}$=(1.2, 1.2) being the easier one to converge from and $\overline{x_{(1)}}$=(-1.2, 1) being the harder one.

Using as starting point $\overline{x_{(0)}} = (1.2, 1.2)$, all the tree methods converge to the actual minimum x^* with an acceptable error.

With the other point $\overline{x_{(1)}}$, instead, both Steepest Descent and Fletcher and Reeves perform well, while Inexact Newton Method doesn't converge, reaching the point (-1.04, 1.07) that is far from x^* . This happens because the hessian computed by the finite differences is not positive definite, thus the method could not converge.

The norm of the gradient of f(x_k) and the error norm, defined as $\epsilon_k = ||x^* - x_k||$, are reported in the tables I, II, III.

TABLE I
RESULTS FOR ROSENBROCK PROBLEM WITH STEEPEST DESCENT

	Steepest Descent Method	
	ϵ_k	$\|\|\nabla f(x_k)\|\|$
$\overline{x_{(0)}} = (1.2, 1.2)$	$1.7595 \cdot 10^{-6}$	$9.9876 \cdot 10^{-7}$
$\overline{x_{(1)}} = (-1.2, 1.0)$	$1.7503 \cdot 10^{-6}$	$9.8334 \cdot 10^{-7}$

TABLE II
RESULTS FOR ROSENBROCK PROBLEM WITH FLETCHER AND REEVES

	Fletcher and Reeves' Method	
	ϵ_k	$\|\|\nabla f(x_k)\|\|$
$\overline{x_{(0)}} = (1.2, 1.2)$	$2.4010 \cdot 10^{-6}$	$9.6493 \cdot 10^{-7}$
$\overline{x_{(1)}} = (-1.2, 1.0)$	$4.4062 \cdot 10^{-7}$	$2.2172 \cdot 10^{-7}$

TABLE III
RESULTS FOR ROSENBROCK PROBLEM WITH INM

	Inexact Newton Method	
	ϵ_k	$\|\|\nabla f(x_k)\|\|$
$\overline{x_{(0)}} = (1.2, 1.2)$	$2.9503 \cdot 10^{-4}$	$1.1781 \cdot 10^{-4}$
$\overline{x_{(1)}} = (-1.2, 1.0)$	2.0321	1.9487

III. GENERALIZED BROYDEN TRIDIAGONAL FUNCTION [P5]

A. Problem definition

$$F(x) = \sum_{i=1}^{n} |(3 - 2x_i)\, x_i - x_{i-1} - x_{i+1} + 1|^p$$
$$p = 7/3, \quad \overline{x_0} = \overline{x_{n+1}} = 0$$
$$\overline{x_i} = -1, \quad i \geq 1.$$

B. Results and discussion

As shown in table IV both SD and CG-FR methods converge as their $||\nabla f(x_k)||$ is lower than the tolerance set, while the IN method fails to archive convergence (it was stopped at 13 iterations as it didn't improve the $||\nabla f(x_k)||$ more than 10^{-12} for 5 consecutive iterations).

Upon further inspection, though, we saw that x_k^{INM} was relatively close to x_K^{SDM} ($||x_k^{SD} - x_k^{INM}|| \approx 3$). By changing the forcing terms to a more aggressive one ($\eta_k = min(0.5, ||\nabla f(x_k)||)$) the method doesn't improve its result and the norm difference remains the same. The CG-FR, instead, converges to a clearly different minimum point ($||x_k^{SD} - x_k^{CG-FR}|| \approx 47$). This would explain why it requires significantly more iterations.

By changing the starting point to the origin ($\overline{x} = 0$) we find out that $||x_k^{SD,0} - x_k^{CG-FR,0}|| \approx 4.5$, so now they are closer, but the fact that the difference was not equal to $||x_k^{SD} - x_k^{INM}||$ led us thinking that the problem had many minimum points. A comparison between $||x_k^{INM,0} - x_k^{CG-FR,0}||$ was not possible as the hessian matrix computed by the IN method in the origin was not positive definite, thus it did not converge. This shows how the starting point is decisive for the convergence of the algorithm.

One other aspect to take into consideration is execution time, as INM requires *significantly* more time to converge compared to the others (about 2 hours against less than one second for SD and ten seconds for CG-FR). It is to be expected, as INM, differently from the other methods, has to compute both $\nabla f \in \mathbb{R}^n$ (the gradient) and $\nabla^2 f \in \mathbb{R}^{n,n}$ (the Hessian) at every iteration. This results computationally very expensive thus requires a lot of time in order to be generated.

TABLE IV
RESULTS FOR P5 STARTING FROM $\overline{x}$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	$9.85 \cdot 10^{-13}$	$8.53 \cdot 10^{-9}$	$6.91 \cdot 10^{-7}$
$\|\|\nabla f(x_k)\|\|$	$8.47 \cdot 10^{-9}$	$9.98 \cdot 10^{-7}$	$1.30 \cdot 10^{-3}$
Iterations (k)	51	3921	13
Time (in seconds)	0.21	10.07	7254

IV. GENERALIZATION OF THE BROWN FUNCTION 2 [P13]

A. Problem definition

$$F(x) = \sum_{j=1}^{k} \left[\left(x_{i-1}^2\right)^{\left(x_i^2+1\right)} + \left(x_i^2\right)^{\left(x_{i-1}^2+1\right)} \right],$$
$$i = 2j, \quad k = n/2,$$
$$\overline{x_i} = -1, \mathrm{mod}(i,2) = 1, \quad \overline{x_i} = 1, \mathrm{mod}(i,2) = 0.$$

B. Results and discussion

TABLE V
RESULTS FOR P13 STARTING FROM $\overline{x}$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	$1.098 \cdot 10^{-13}$	$1.95 \cdot 10^{-13}$	$1.01 \cdot 10^{-24}$
$\|\|\nabla f(x_k)\|\|$	$6.62 \cdot 10^{-7}$	$8.83 \cdot 10^{-7}$	$2.01 \cdot 10^{-12}$
Iterations (k)	15	10	7
Time (in seconds)	0.025	0.026	3624

As shown in table V all methods converged. This problem shows well enough the convergence rates of the methods used: the fastest was the IN method with 7 iterations. It was to be expected as it as its convergence rate is superlinear. The slowest was the SD method having done more than double the iterations (15 iterations) which is understandable as the convergence rate is only linear. The CG-FR method, instead, finished in 10 iterations. It shows how CG-FR method is an improvement of SD method and its convergence rate is superlinear, yet slower than IN method.

For this problem all methods converged to the same minimum, even if with different precisions. The most precise was the IN method as its $||\nabla f^{INM}||$ had the order of magnitude of 10^{-24} it came very close to a real minimum. The other methods had $||\nabla f||$ orders of maginitude of about 10^{-13}, which is also pretty accurate, but less accurate than the IN method. The precision gained with the IN method is paid, though, in computational time: for the reasons explained in the discussion of P5, the computation of the Hessian and of the gradient of the function at each iteration require a lot of time to be generated.

V. Discrete boundary value problem [P14]

A. Problem definition

$$F(x) = \sum_{i=1}^{n} \left[2x_i - x_{i-1} - x_{i+1} + h^2 \left(x_i + ih + 1 \right)^3 / 2 \right]^2,$$
$$h = 1/(n+1), \quad \overline{x_0} = \overline{x_{n+1}} = 0,$$
$$\overline{x}_i = ih(1-ih), \quad i \geq 1.$$

B. Results and discussion

As shown in table VI this problem, with starting point $\overline{x}$, does not converge with any of the tree methods analyzed.
In particular, INM stops after only 5 iterations since it does not improve the $||\nabla f(x_k)||$ more than 10^{-12} for 5 consecutive iterations, even changing forcing terms.

Inspecting deeper this method, we have seen that at each iteration the Hessian is non Positive-Definite and in this case it's common for the INM to not converge since it is not a completely reliable method.

After this consideration, we have tried to change starting point, using the origin ($\overline{x} = 0$) and obtaining the results reported in table VII. Even with 10^5 iterations, Steepest Descent Method and Fletcher & Reeves don't converge, while Inexact Newton Method reaches the desired tolerance.

Furthermore, we saw that $||x_k^{INM,0} - x_k^{SD,0}|| \approx 7.3 \cdot 10^{-4}$ and $||x_k^{CG-FR,0} - x_k^{SD,0}|| \approx 1.3 \cdot 10^{-3}$, so the last points computed by the methods are very close.

TABLE VI
Results for P14 starting from $\overline{x}$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	$2.1639 \cdot 10^{-8}$	$2.1614 \cdot 10^{-8}$	$2.1727 \cdot 10^{-8}$
$\|\|\nabla f(x_k)\|\|$	$1.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$	$1.2974 \cdot 10^{-5}$
Iterations (k)	50000	50000	5
Time (in seconds)	1149	1053	3216

TABLE VII
Results for P14 starting from $\overline{x} = 0$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	$4.4273 \cdot 10^{-9}$	$4.3919 \cdot 10^{-9}$	$2.1727 \cdot 10^{-8}$
$\|\|\nabla f(x_k)\|\|$	$6.3563 \cdot 10^{-4}$	$6.3167 \cdot 10^{-4}$	$9.9263 \cdot 10^{-7}$
Iterations (k)	100000	100000	7
Time (in seconds)	1996	1463	6807

VI. Banded trigonometric problem [P16]

A. Problem definition

$$F(x) = \sum_{i=1}^{n} i \left[(1 - \cos x_i) + \sin x_{i-1} - \sin x_{i+1} \right]$$
$$x_0 = x_{n+1} = 0 \qquad \overline{x}_i = 1, \quad i \geq 1$$

B. Results and discussion

The methods taken in consideration don't converge with the starting point $\overline{x}$, , even if they all reach the $f(x_k)$, as reported in table VIII.

By inspecting INM, we have seen in the first ~ 8 iterations the Hessian is not positive-definite, but after that it becomes positive-definite. Despite this change, the method does not converge and also takes lots of time. We stopped after 50 iterations as

Using as starting point $\overline{x} = 0$ (and increasing the maximum number of iterations), we observed slightly better results, as reported in table IX. Among the tree methods, only steepest descent converges.

The interesting thing to notice is that with both starting points and all the methods, the value $f(x_k)$ is the same.

In order to assure that the non-convergence is an issue given by the size of the problem we conducted a further trial changing the size from $n = 1000$ to $n = 100$. Indeed, for this problem all methods converged, so we understand that with enough time and iterations, also the n=1000 problem converges. Results are shown in table X.

TABLE VIII
Results for P16 starting from $\overline{x}$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	-427.4045	-427.4045	-427.4041
$\|\|\nabla f(x_k)\|\|$	$1.2157 \cdot 10^{-5}$	$7.3911 \cdot 10^{-5}$	0.0002
Iterations (k)	50000	50000	50
Time (in seconds)	94	159	6185

TABLE IX
Results for P16 starting from $\overline{x} = 0$

	SD Method	CG-FR Method	IN Method
$f(x_k)$	-427.4045	-427.4045	-427.4045
$\|\|\nabla f(x_k)\|\|$	$9.4604 \cdot 10^{-6}$	$1.3258 \cdot 10^{-4}$	$6.9677 \cdot 10^{-5}$
Iterations (k)	50000	50000	100
Time (in seconds)	174	230	16347

TABLE X
Results for P16 starting from $\overline{x}$ with n=100

	SD Method	CG-FR Method	IN Method
$f(x_k)$	-49.99	-49.99	-49.99
$\|\|\nabla f(x_k)\|\|$	$9.6638 \cdot 10^{-7}$	$9.3319 \cdot 10^{-7}$	$4.4363 \cdot 10^{-6}$
Iterations (k)	210	115	150
Time (in seconds)	0.36	0.05	41

VII. Conclusions

As shown in the previous paragraphs, each problem has its own unique issues. Based on the problem faced each method gives different results.

We saw that the starting point plays a crucial role in the convergence of the methods. In particular the INM could compute a non positive-definite hessian, thus risking to never converge. Even if it arrives at convergence, x_k^{INM} can be different than x_k^{SD} or x_k^{CG-FR}. We can conclude that the INM is not reliable in such a situation. By changing the starting point we saw that this problem can be avoided.

Also the size of the problem can increase significantly the computational time. In particular for the INM, at each iteration there are $\sim 4 \cdot 10^6$ evaluations of the function $f(x)$ for $n = 1000$ for the Hessian alone. This issue does not arise for SD and CG-FR since they need the exact gradient (and they don't need the Hessian) and they converge very fast, even if more iterations are needed.

Generally, if none of the previous problems arise, we can say that the theoretical considerations about convergence rates of the methods hold. The INM more often is the best choice, being both the more precise and the one that performs less iterations. The computational time, though, can be very high if finite differences are used.

References

[1] https://www.researchgate.net/publication/325314497_Test_Problems

[2] "A comparative study of conjugate gradient and steepest descent methods for optimization" by L. Sonneveld, Journal of Global Optimization, Volume 42, Issue 2, pp. 505-525, 2008.