
Second Order Momentum

Vaibhav Sinha

cs15btech11034@iith.ac.in

Sukrut Rao

cs15btech11036@iith.ac.in

Abstract

We study the effect of momentum on two second order methods - Newton's method and BFGS. We empirically determine the performance as compared to gradient descent, with and without momentum, as well as Cubic Regularization. For applying momentum, we consider both the classical momentum and Nesterov's accelerated gradient.

1 Objective and Introduction

To achieve faster convergence, second order methods were proposed over first order methods. Although second order methods have better theoretical bounds than first order methods and converge in very few steps, they are often not used in practice due to the cost of each step. To cope with these problems, two approaches have been proposed and well studied: (i) Adding momentum [1, 2] to first order methods to make the convergence speed comparable to second order methods, (ii) Quasi-Newton methods [3–5] which avoid the costly computations involved in second order methods, with a small cost on the rate of convergence. In this project, we aim to combine both these approaches and attempt to verify if the classical second order methods as well as Quasi-Newton methods can be accelerated.

Momentum has been extensively studied on first order methods [1, 2]. However, there seems to be almost no work done on the effects of momentum on Newton's method. Through this project we seek to address this problem. [?] proposes acceleration of cubic regularization but this acceleration is not an extension to classical or nesterov's momentum and hence can not be directly applied to Newton's method or BFGS.

We apply momentum to two methods - Newton's method and BFGS. We use both the classical momentum proposed by [1] and Nesterov's accelerated gradient proposed by Sutskever et al. [2]. A comparison of these approaches is made with gradient descent, gradient descent with momentum, and for some functions, with Cubic Regularization. We study the convergence rates empirically for three convex and four non-convex functions, mostly taken from various benchmarks.

The report is organized as follows. Section 2 describes the two momentum methods and their update formulae. In Section 3, we describe our extension of the formulation of momentum for second order methods. Then, in Section 4, we describe the methods used for optimization, such as gradient descent and BFGS. This is followed by description of the methods used for finding step sizes in Section 5. We list the functions used to empirically test our approach in Section 6. The experiments conducted then follow in Section 7, and their results are described in Section 8. Based on these results, we provide our inferences subsequently in Section 9, and conclude in Section 10.

The code used in this project can be found in its GitHub repository at <https://github.com/vbsinha/Second-Order-Momentum>.

2 Momentum

The objective of momentum methods is to accelerate the descent towards the optimal value. It 'damps' the gradient descent when close to the optimal, that allows the use of higher learning rates while avoiding overshooting the point of minimum.

Momentum was first proposed by [1]. Subsequently, in 2013, usage of Nesterov's accelerated gradient for deep learning was proposed by Sutskever et al. [2], which gave good empirical results.

The update step for gradient descent is given by

$$x^+ = x - \eta \nabla f(x) \quad (1)$$

In general, with an optimization method that updates by U , the update step is given by

$$x^+ = x - \eta U \quad (2)$$

With momentum, this update step is modified to include a velocity term, and the update depends on the relative weights given to the velocity direction and the descent direction, parameterized by a momentum parameter γ .

2.1 Classical Momentum

In this method, we introduce a velocity v , which is initialized with 0. Then, the update step is given by

$$v^+ = \gamma v + (1 - \gamma) \nabla f(x) \quad (3)$$

$$x^+ = x - \eta v^+ \quad (4)$$

This will be referred to as just 'momentum' subsequently in this report, when clear from the context.

2.2 Nesterov's accelerated gradient

In this method, we have γ and v as before, and the update is given by

$$v^+ = \gamma v + (1 - \gamma)x \quad (5)$$

$$x^+ = v^+ - \eta \nabla f(v^+) \quad (6)$$

This will be referred to as 'Nesterov momentum' subsequently in this report, when clear from the context.

3 Momentum for Second Order Methods

The original idea of momentum follows from the rolling ball analogy. The intuition behind momentum is that along with the direction of the current gradient we also descend in the direction by which we reached at the current position. This helps since the lower we descend, gradient becomes smaller and so moving extra in the previous direction helps us converge to the minima faster. This intuition is confirmed by the mathematical analysis of convergence which proves faster convergence with momentum for first order method.

We can follow the same intuition of momentum for first order method and ask if adding momentum will help in converging faster for second order methods. To examine this we propose a simple and straightforward extension of the classical momentum and Nesterov's accelerated gradient for second order methods. We replace the $\nabla f(x)$ used in the existing formulations by $U(x)$, where U is the update of the method under consideration. As described briefly in 4, $U(x) = (\nabla^2 f(x))^{-1} \nabla f(x)$ for Newton's method and $U(x) = H \nabla f(x)$ for BFGS.

The classical momentum update is now given by

$$v^+ = \gamma v + (1 - \gamma)U(x)$$

$$x^+ = x - tv^+$$

Similarly, the Nesterov momentum update is now given by

$$v^+ = \gamma v + (1 - \gamma)x \quad (7)$$

$$x^+ = v^+ - \eta U(v^+) \quad (8)$$

4 Optimization Methods

This section describes the optimization methods used in this project.

4.1 Gradient Descent

This is a first order method, which updates the position x to a scalar multiple of the direction of the steepest descent, given by the gradient of the function at x . As shown before, the update is given by

$$x^+ = x - \eta \nabla f(x) \quad (9)$$

where $\eta > 0$ is a (small) constant.

4.2 Newton's Method

This is a second order method. The motivation behind this is to include the second order term in the Taylor approximation of the function to obtain the next step. The update is given by

$$x^+ = x - \eta (\nabla^2 f(x))^{-1} \nabla f(x) \quad (10)$$

A drawback of this method is that computing the inverse of the Hessian can be costly.

4.3 Broyden-Fletcher-Goldfarb-Shanno (BFGS) Update

This is a Quasi-Newton method. This provides a technique to approximate the inverse of the Hessian as used in Newton's method, with an objective of reducing the cost of computation.

Let

$$s = x^+ - x \quad (11)$$

$$y = \nabla f(x^+) - \nabla f(x) \quad (12)$$

Then, the approximate of the inverse Hessian, H , is updated as follows

$$H^+ = \left(I - \frac{sy^T}{y^T s} \right) H \left(I - \frac{ys^T}{y^T s} \right) + \frac{ss^T}{y^T s} \quad (13)$$

The update is then given by

$$x^+ = x - \eta H \nabla f(x) \quad (14)$$

4.4 Cubic Regularization

This was proposed in [6]. Cubic Regularization builds up on Netwon's method. Netwon's method obtains x^+ as:

$$x^+ = \operatorname{argmin}_y f(x) + \nabla f(x)^T(y - x) + \frac{1}{2}(y - x)^T \nabla^2(y - x) \quad (15)$$

Cubic regularization goes one term forward in the Taylor expansion and obtains x^+ as:

$$x^+ = \operatorname{argmin}_y f(x) + \nabla f(x)^T(y - x) + \frac{1}{2}(y - x)^T \nabla^2(y - x) + \frac{L}{6} \|y - x\|_2^3 \quad (16)$$

5 Step Sizes

The choice of step size can greatly influence the rate of convergence. In our project, we consider two choices - using a fixed step size, and the popular step size found using backtracking line search. The step size is given by η , as used in the equations previously.

5.1 Fixed Step

In this, the value of η is fixed. While it is a simple choice, it needs to be carefully chosen. A very small value will take a very large number of steps to converge, and a very large value will cause the step to overshoot the minimum, and sometimes result in divergence.

5.2 Backtracking Line Step

This is a method to adaptively select the step size. It is parameterized by α, β , where $0 < \alpha \leq 0.5$ and $0 < \beta < 1$. The step size is found iteratively. Initially, the step size η is set to 1. Then, while

$$f(x - \eta U) > f(x) - \alpha \eta \nabla f(x)^T U \quad (17)$$

we update $\eta = \beta \eta$. Here, U is the update based on the method used.

This works well and is useful for computationally intensive methods like Newton's method, as it reduces the number of iterations needed.

When using momentum, we replace x and U with values as prescribed by momentum update. So, in the case of classical momentum, we use x and v^+ , and in the case of Nesterov momentum, we use v^+ and $U(v^+)$.

It can be observed that we perform backtracking line search on the momentum update, and not the regular update, to find η . This is because if we use the regular update, the value of η obtained will be a 'good' value assuming we are descending based on the regular update. Since the update includes a momentum term, we perform backtracking line search taking that into account. Nevertheless, we performed experiments in both cases - (i) performing backtracking line search on the update after momentum, and (ii) performing backtracking line search on the update before momentum. We observed across all functions and both momentum types, (i) performed significantly better than (ii), and hence we use that in our study.

6 Functions Used

We now list the functions used to empirically test our approach. This consists of three convex and four non-convex functions, most of which are taken from benchmark functions for optimization problems.

6.1 Non-convex functions

6.1.1 Beale Function

$$f(x, y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2 \quad (18)$$

where $-4.5 \leq x, y \leq 4.5$, $f(x^*) = 0$.

6.1.2 Bird Function

$$f(x, y) = \sin(x)e^{(1-\cos(y))^2} + \cos(y)e^{(1-\sin(x))^2} + (x - y)^2 \quad (19)$$

where $-2\pi \leq x, y \leq 2\pi$, $f(x^*) = -106.764537$.

6.1.3 Rosenbrock Function

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2 \quad (20)$$

where $f(x^*) = 0$ at $x^* = (1, 1)$.

6.1.4 Levy13 Function

$$f(x, y) = \sin^2(3\pi x) + (x - 1)^2 (1 + \sin^2(3\pi y)) + (y - 1)^2 (1 + \sin^2(2\pi y)) \quad (21)$$

where $f(x^*) = 0$.

6.2 Convex functions

6.2.1 Quadratic function for cubic regularization (Cubic Reg function)

$$f(x, y) = x^2 + y^2 + x^2y^2 \quad (22)$$

where $f(x^*) = 0$ for $x^* = (0, 0)$

6.2.2 Quadratic Function

$$f(x, y) = 1.125x^2 + 0.5xy + 0.75y^2 + 2x + 2y \quad (23)$$

where $x^* = (-0.64, -1.12)$

6.2.3 Sum of 2 convex functions (Ryan's function)

$$f(x, y) = \frac{10x^2 + y^2}{2} + 5 \log(1 + \exp(-x - y)) \quad (24)$$

where $x^* = (0.11246718, 1.1246718)$

7 Methodology

7.1 Experiments

We performed a range of experiments across methods, step sizes, and functions. The following are the methods we used for optimizing each function:

1. Gradient Descent (First Order)
2. Gradient Descent with momentum
3. Gradient Descent with Nesterov momentum
4. Newton's method (Second Order)

- 5. Newton's method with momentum
- 6. Newton's method with Nesterov momentum
- 7. BFGS (Quasi-Newton method)
- 8. BFGS with momentum
- 9. BFGS with Nesterov momentum
- 10. Cubic Regularization (only for some functions)

We use Cubic Regularization only on some functions as it requires that the Hessian of the function be Lipschitz, with a known Lipschitz constant L .

We separately test on both fixed and backtracking step size. We choose the values of parameters for the step sizes for each method such that, roughly, they provide the best observed convergence rates.

7.2 Performance Metrics

A well performing optimization method is characterized by a high rate of convergence over a wide range of problems. The performance of different techniques can be compared by finding the distance from the optimum after a set of fixed, predefined number of iterations.

In other words, we compute

$$|f(x^{(k)}) - f(x^*)| \quad (25)$$

for fixed values of k and a set of functions f for each class of methods.

8 Results

In this section we show the results that we obtain over the different functions. We begin with convex functions and then show non convex functions.

For each function, contour plots are shown that characterize the paths taken by each method. Log-scale plots of the convergence rates for each of the methods are also provided, where the value of $f - f^*$ is plotted against k . For each function, a table of results shows the values of $f - f^*$ for certain fixed values of k .

8.1 Convex Functions

8.1.1 Quadratic Function

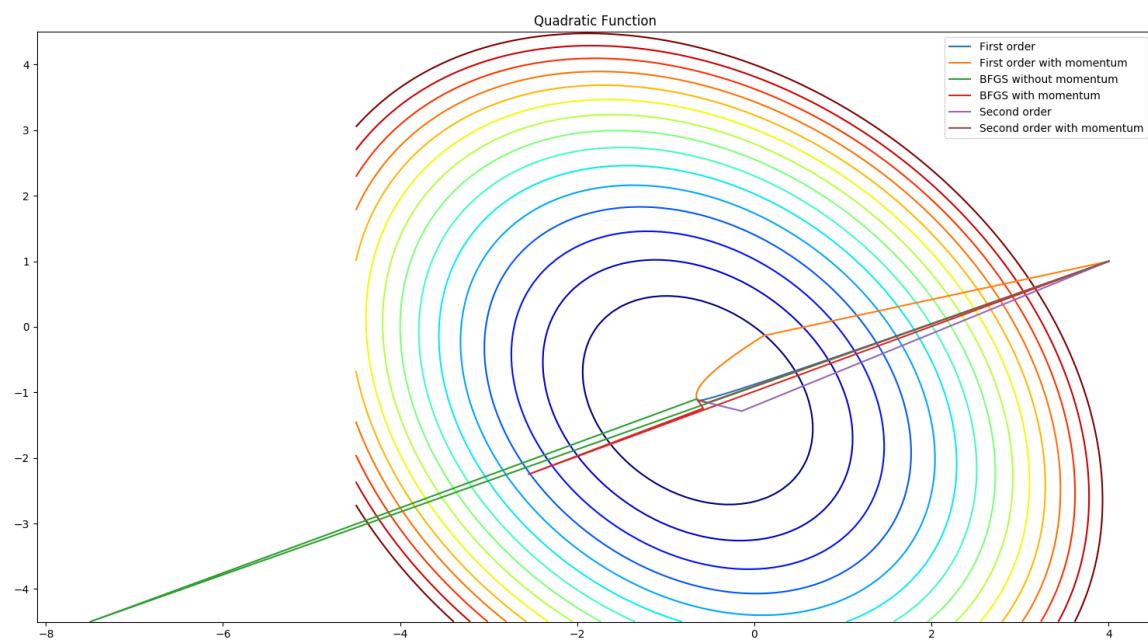


Figure 1: Without Backtracking nesterov momentum for quadratic function

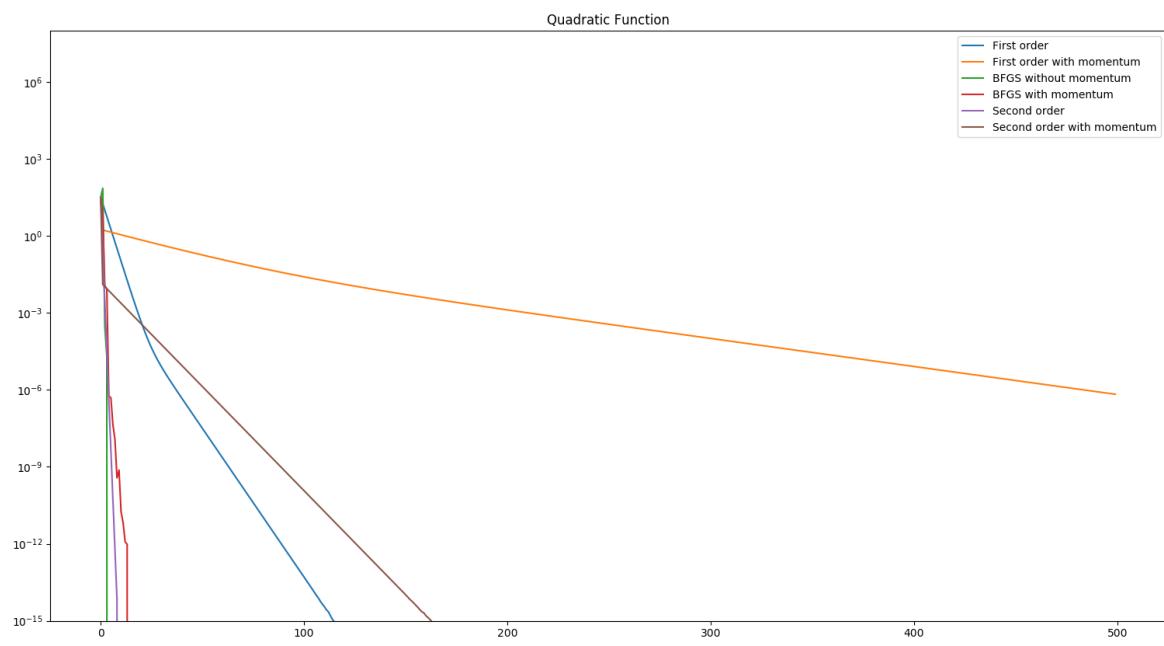


Figure 2: Without Backtracking nesterov momentum convergence for quadratic function

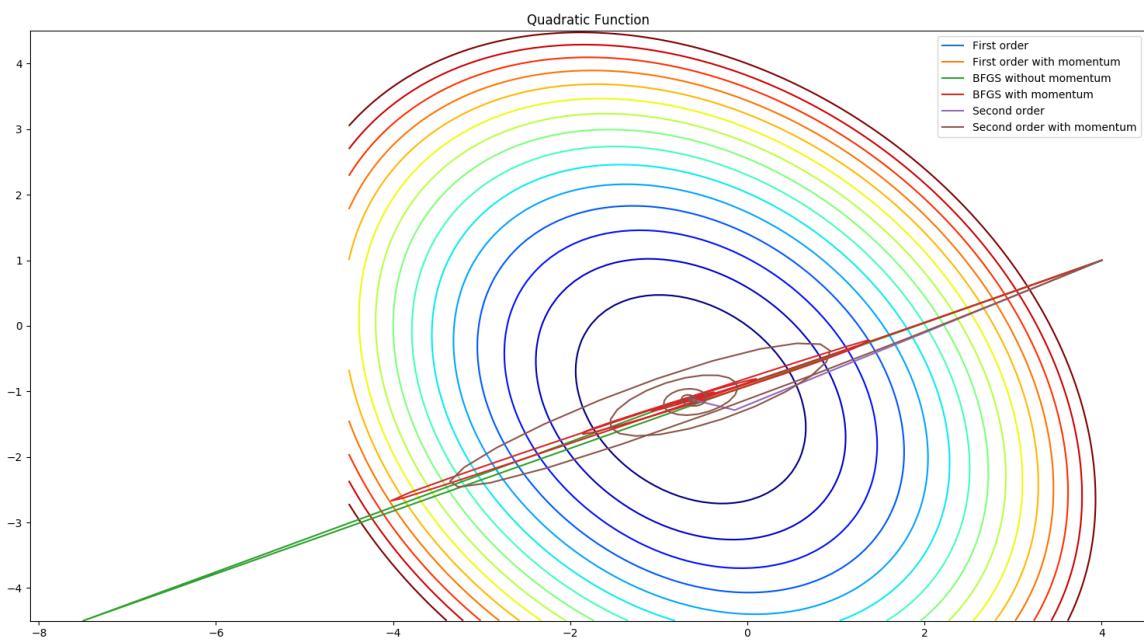


Figure 3: Without Backtracking, classical momentum for quadratic function

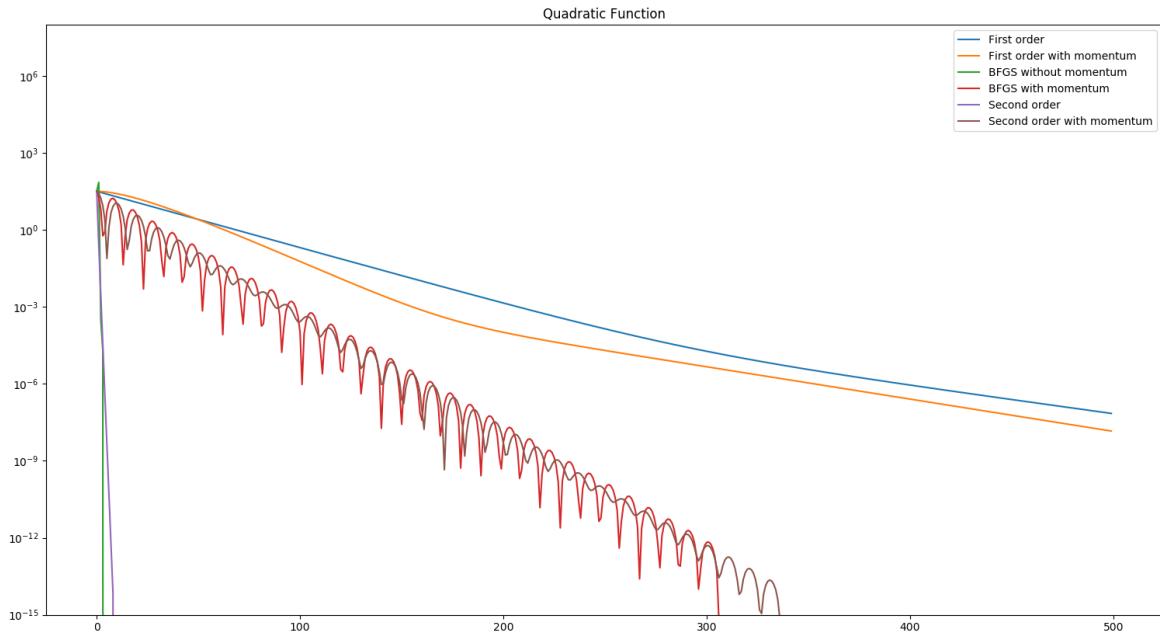


Figure 4: Without Backtracking, classical momentum convergence for quadratic function

Table 1: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	32.51	32.51	32.51	32.51	32.51	32.51	32.51	32.51	32.51
5	1.83	30.42	1.420	4.470e-09	0.0785	0.00612	0.0	5.544	0.0276
10	0.104	26.263	1.120	0.0	11.193	0.00240	0.0	11.270	11.270
20	0.000422	16.8687	0.7014	0.0	3.7783	0.00037	0.0	3.862	3.862
50	3.176e-08	2.55	0.182	0.0	0.128	1.352e-06	0.0	0.144	0.144
100	5.062e-14	0.0598	0.0257	0.0	0.000265	1.173e-10	0.0	0.000454	0.000454
250	8.881e-16	2.0178e-05	0.0003621	0.0	7.121e-11	8.881e-16	0.0	1.219e-11	1.219e-11
500	8.881e-16	1.413e-08	6.625e-07	0.0	6.661e-16	8.881e-16	0.0	4.440e-16	4.440e-16

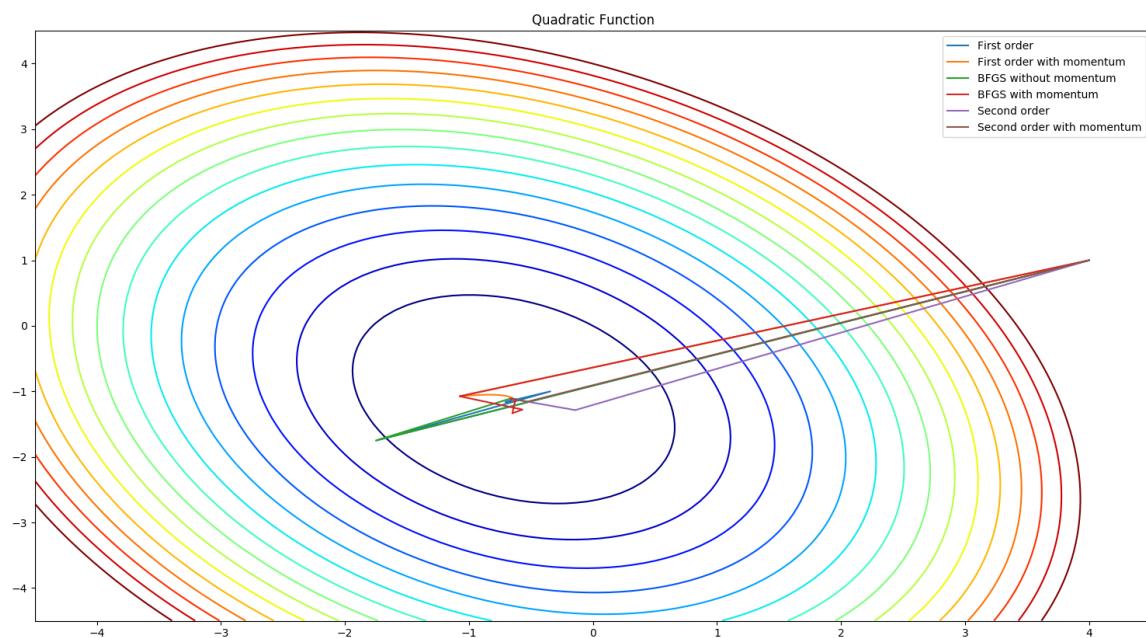


Figure 5: With Backtracking nesterov momentum for quadratic function

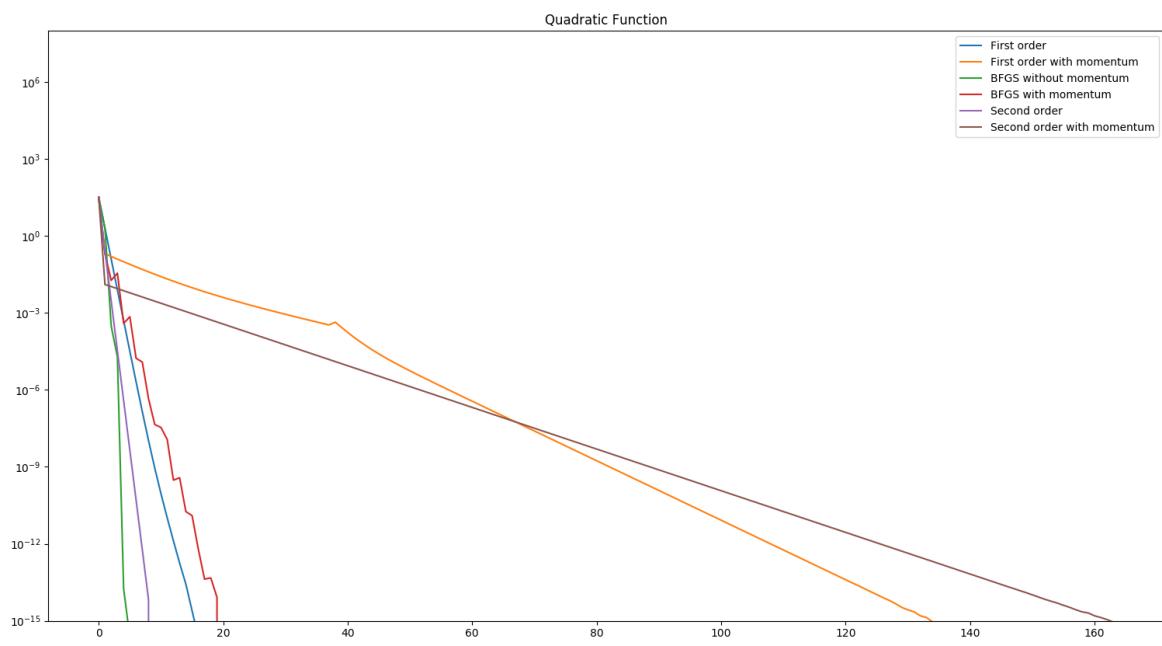


Figure 6: With Backtracking nesterov momentum convergence for quadratic function

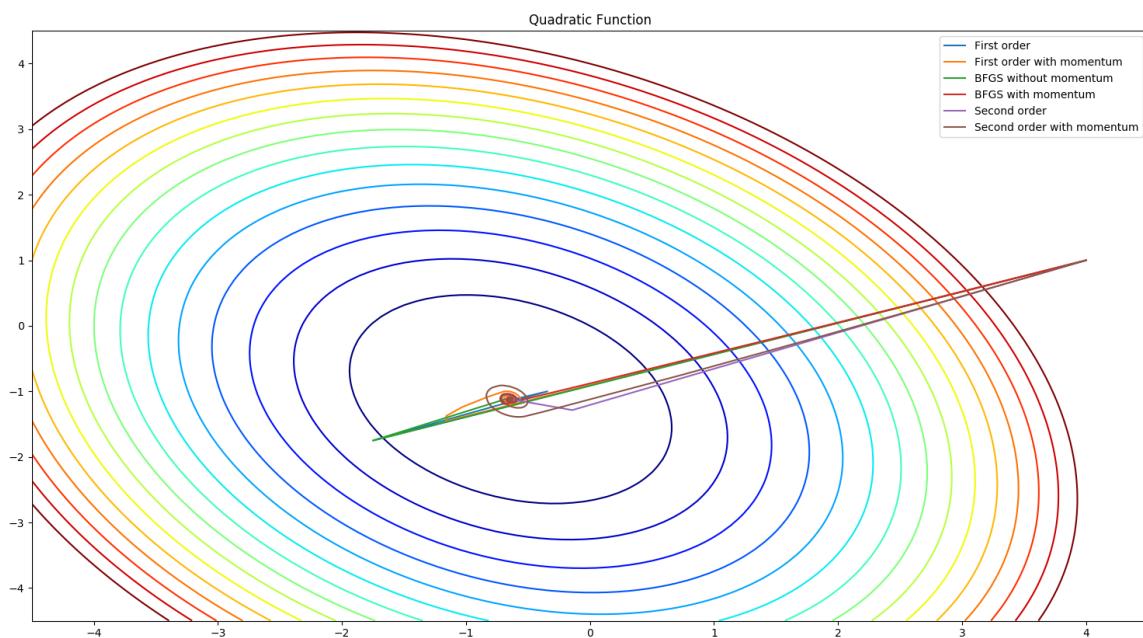


Figure 7: With Backtracking, classical momentum

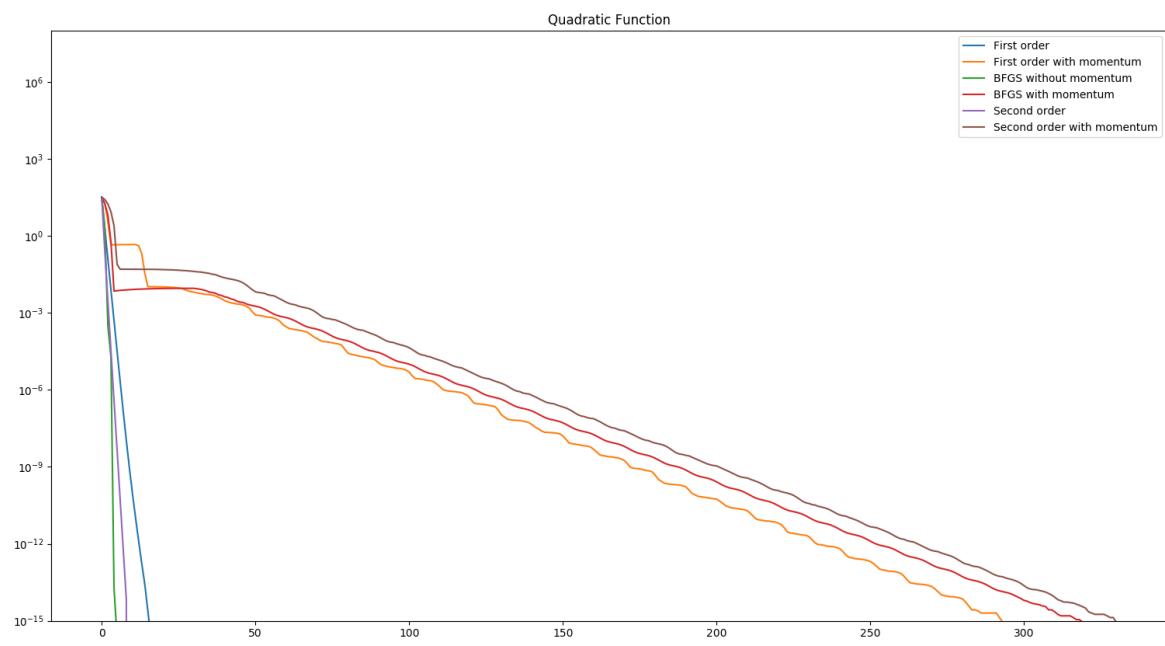


Figure 8: With Backtracking, classical momentum convergence

8.1.2 Quadratic function for Cubic Regularization

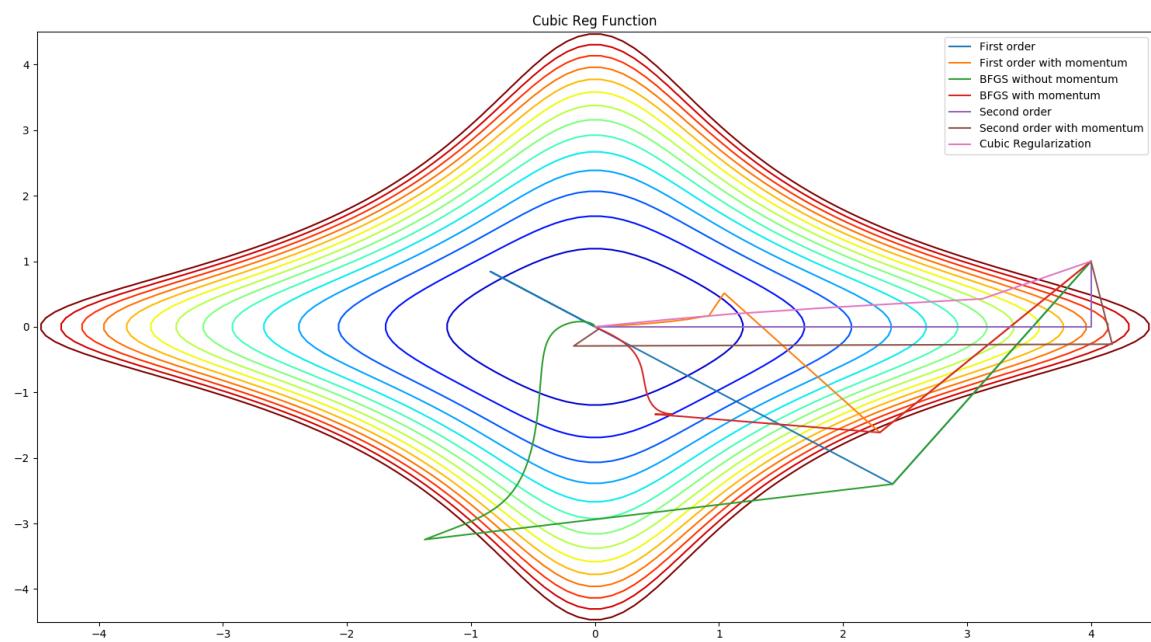


Figure 9: Without Backtracking nesterov momentum for cubic reg function

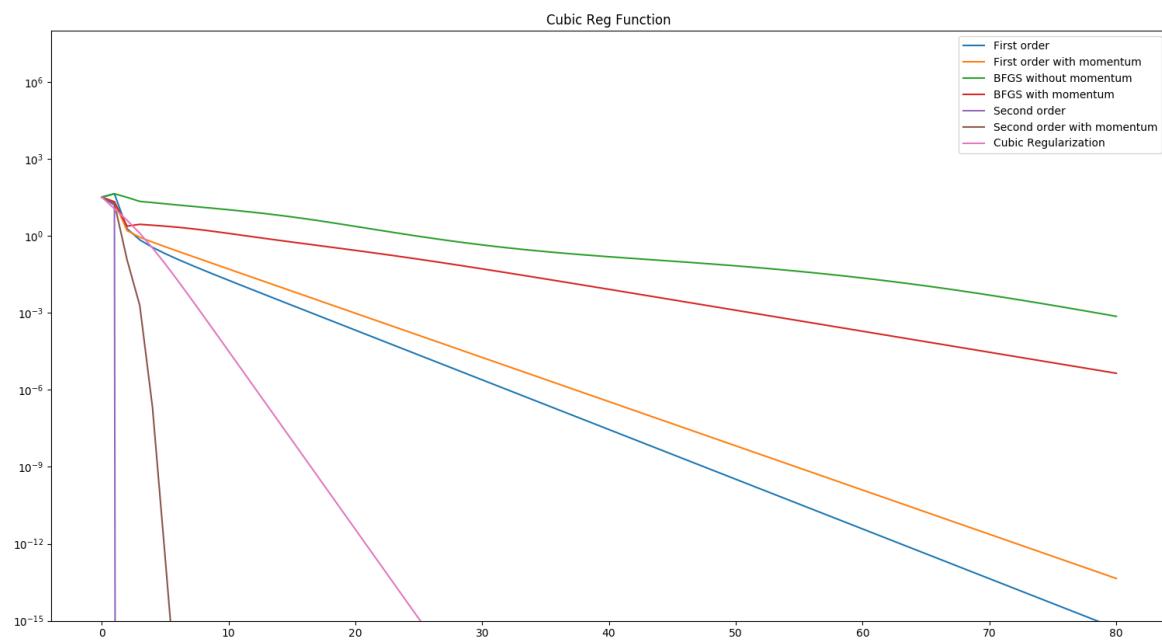


Figure 10: Without Backtracking nesterov momentum convergence for cubic reg function

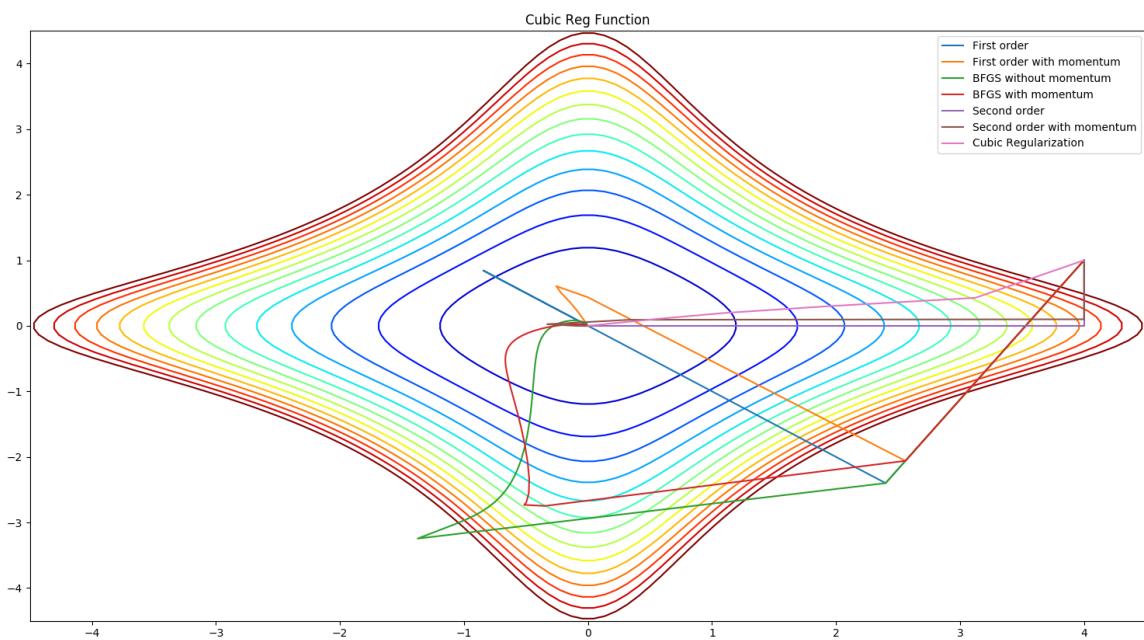


Figure 11: Without Backtracking, classical momentum for cubic reg function

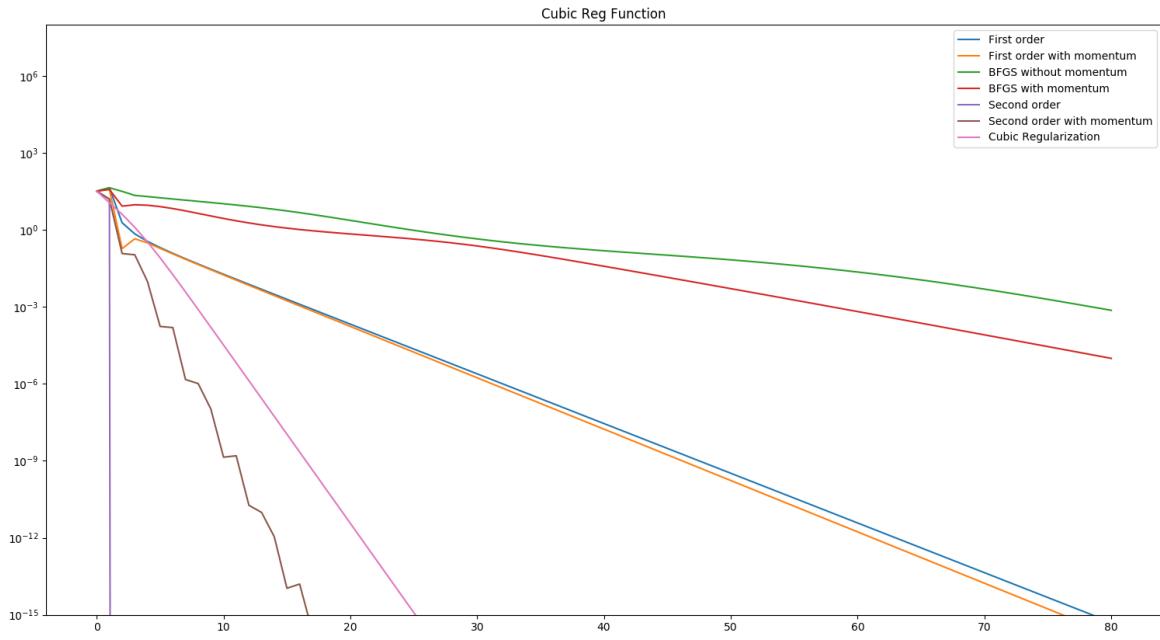


Figure 12: Without Backtracking, classical momentum convergence for cubic reg function

Table 2: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	33	33	33	33	33	33	33	33	33
5	0.207	0.192	0.385	0.0	0.000174	3.0325e-13	18.1426	8.2262	12.672
10	0.0189	0.0177	0.0519	0.0	1.393e-09	3.044e-19	11.661	3.0825	2.956
20	0.000214	0.000174	0.000979	0.0	2.252e-16	3.044e-19	1.3142	2.313	1.723
50	3.285e-10	1.712e-10	6.604e-09	0.0	2.252e-16	3.044e-19	0.474	1.055	0.0199
100	7.866e-16	6.697e-16	8.412e-16	0.0	2.252e-16	3.044e-19	1.794e-05	3.157e-06	2.435e-07
250	7.866e-16	6.697e-16	8.412e-16	0.0	2.252e-16	3.044e-19	9.874e-16	9.036e-16	9.133e-16
500	7.866e-16	6.697e-16	8.412e-16	0.0	2.252e-16	3.044e-19	9.319e-16	9.129e-16	8.936e-16

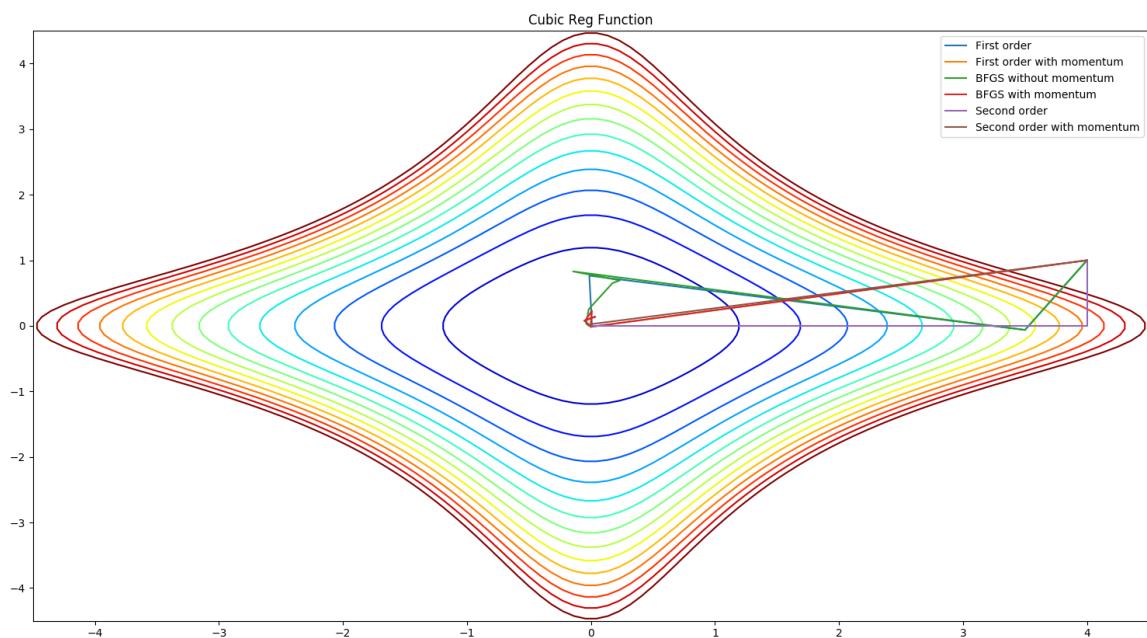


Figure 13: With Backtracking nesterov momentum for cubic reg function

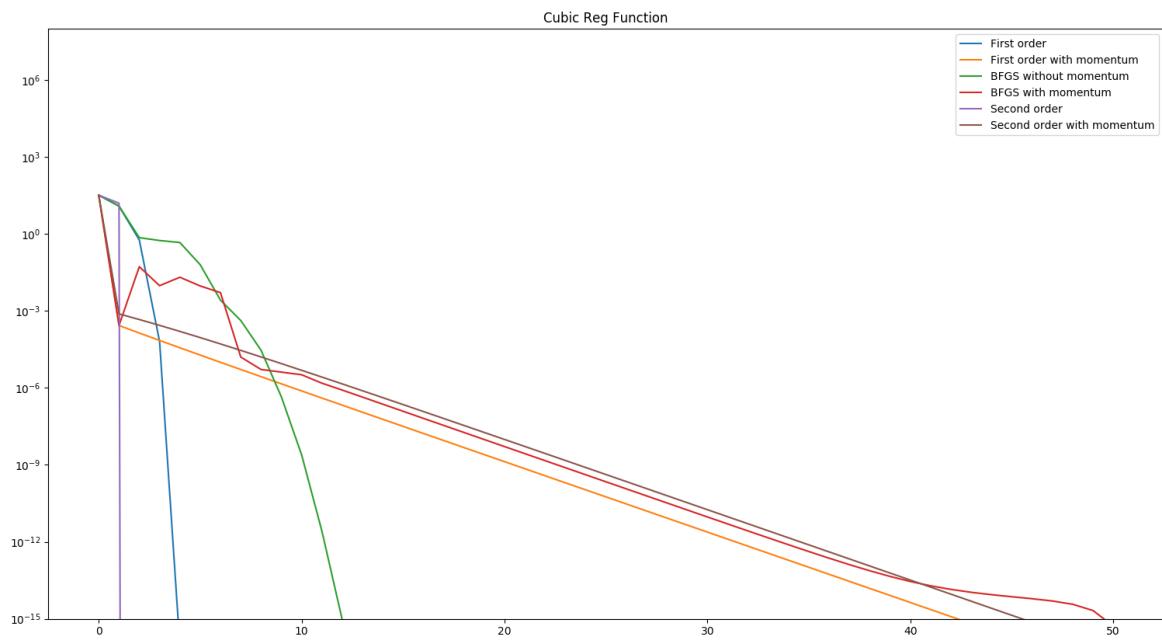


Figure 14: With Backtracking nesterov momentum convergence for cubic reg function

8.1.3 Ryan's function

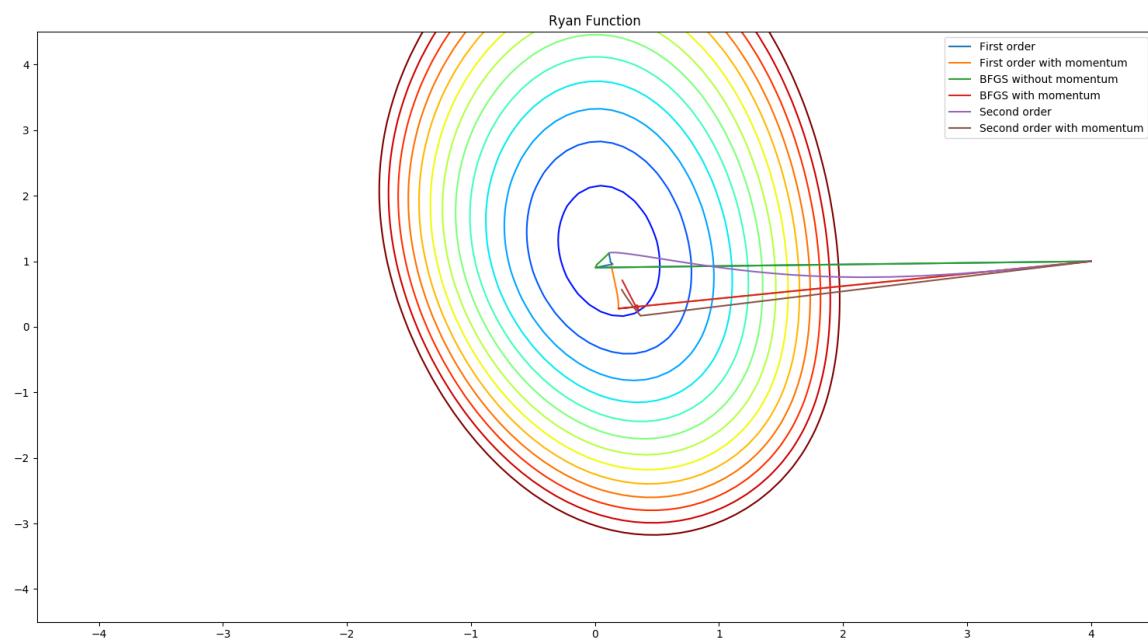


Figure 15: Without Backtracking nesterov momentum for Ryan's function

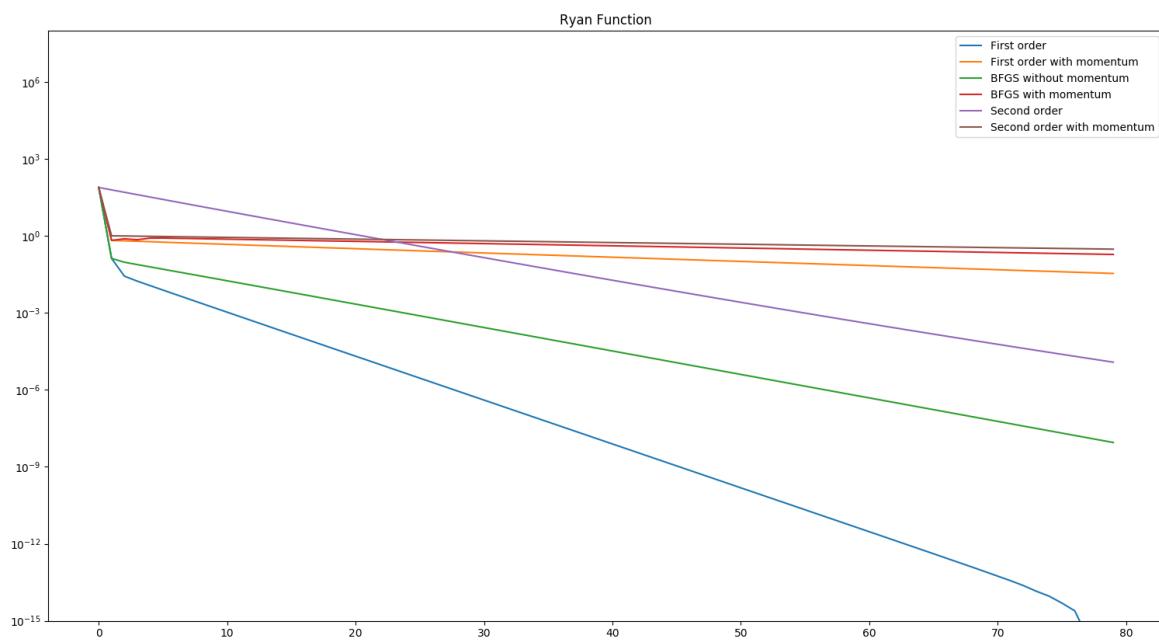


Figure 16: Without Backtracking nesterov momentum convergence for Ryan's function

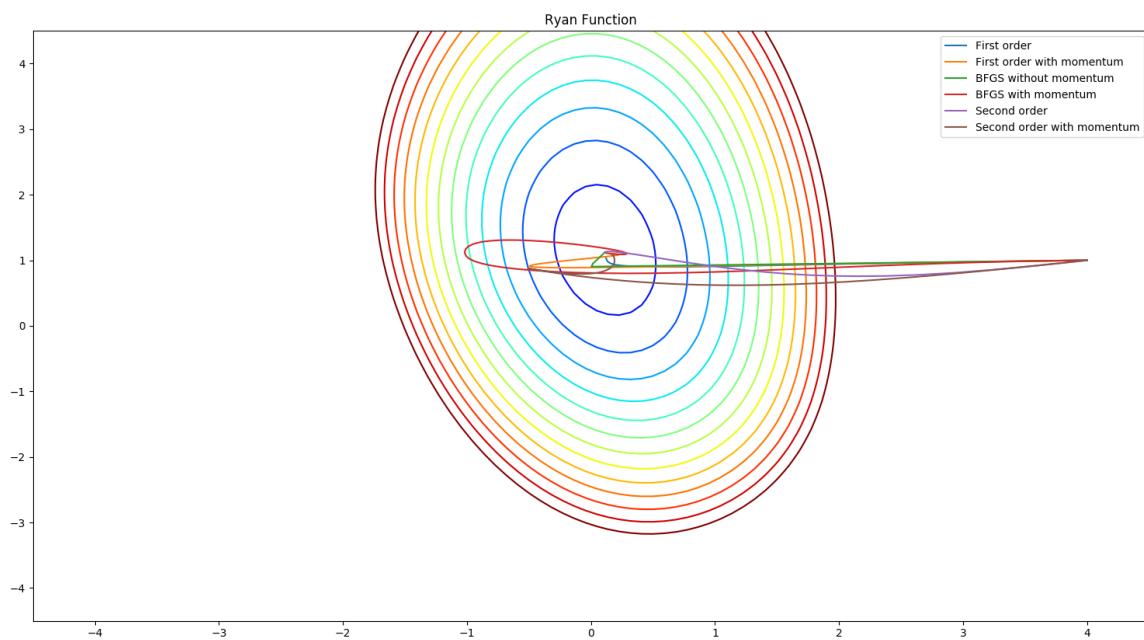


Figure 17: Without Backtracking, classical momentum for Ryan's function

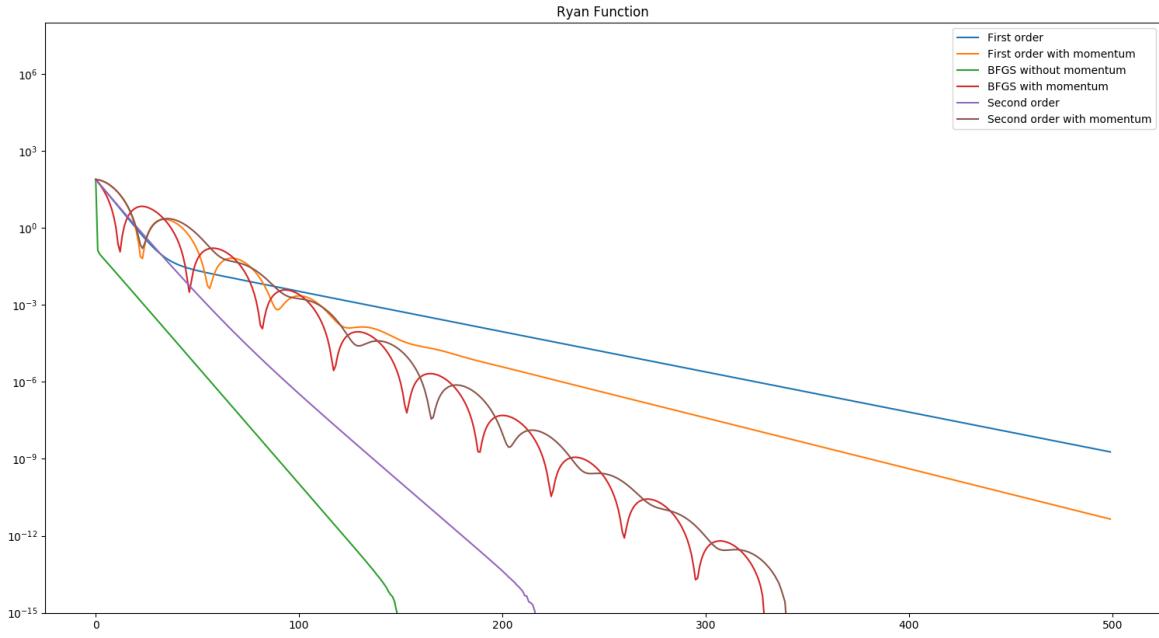


Figure 18: Without Backtracking, classical momentum convergence for Ryan's function

Table 3: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	78.0638	78.0638	78.0638	78.0638	78.0638	78.0638	78.0638	78.0638	78.0638
5	26.638	59.370	1.105	26.830	59.371	0.961	0.0510	20.267	59.473
10	8.837	29.556	1.073	9.247	29.616	0.88	9.191	29.609	29.677
20	0.953	0.740	1.0156	1.136	0.962	0.761	1.106	0.928	0.924
50	0.0219	0.138	0.87197	0.00261	0.562	0.480	0.00198	0.155	0.245
100	0.00340	0.00227	0.696	3.452e-07	0.00176	0.225	5.264e-08	0.000673	0.000699
250	1.494e-05	3.892e-07	0.381	2.220e-16	2.639e-10	0.0236	6.661e-16	3.364e-10	3.366e-10
500	1.803e-09	4.287e-12	0.147	2.220e-16	2.220e-16	0.000531	6.661e-16	-8.881e-16	2.220e-16

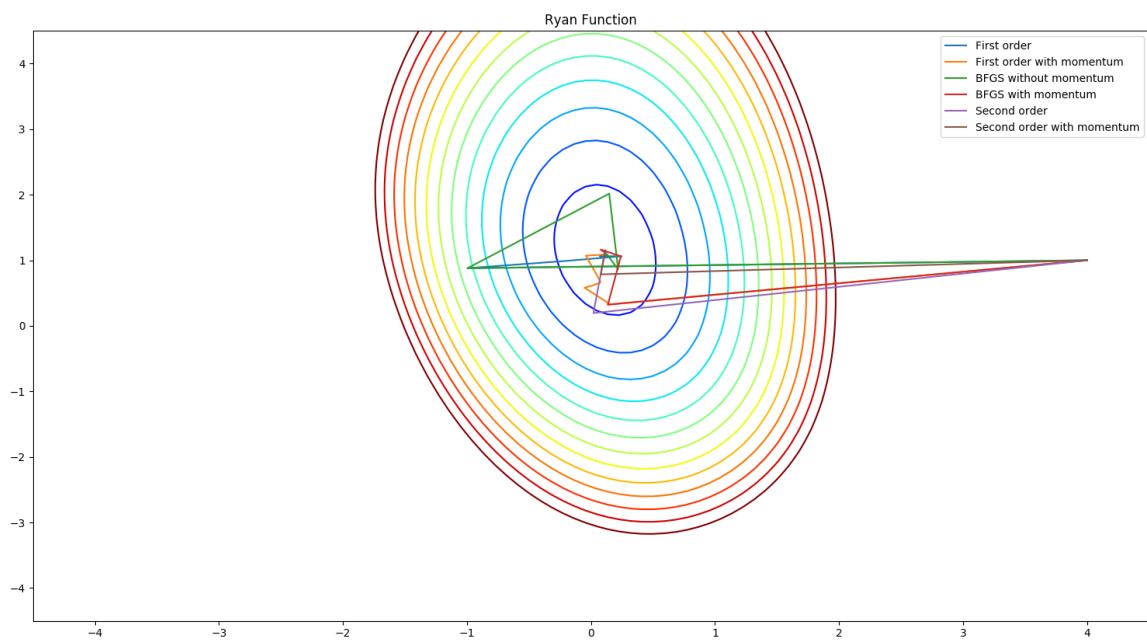


Figure 19: With Backtracking nesterov momentum for Ryan's function

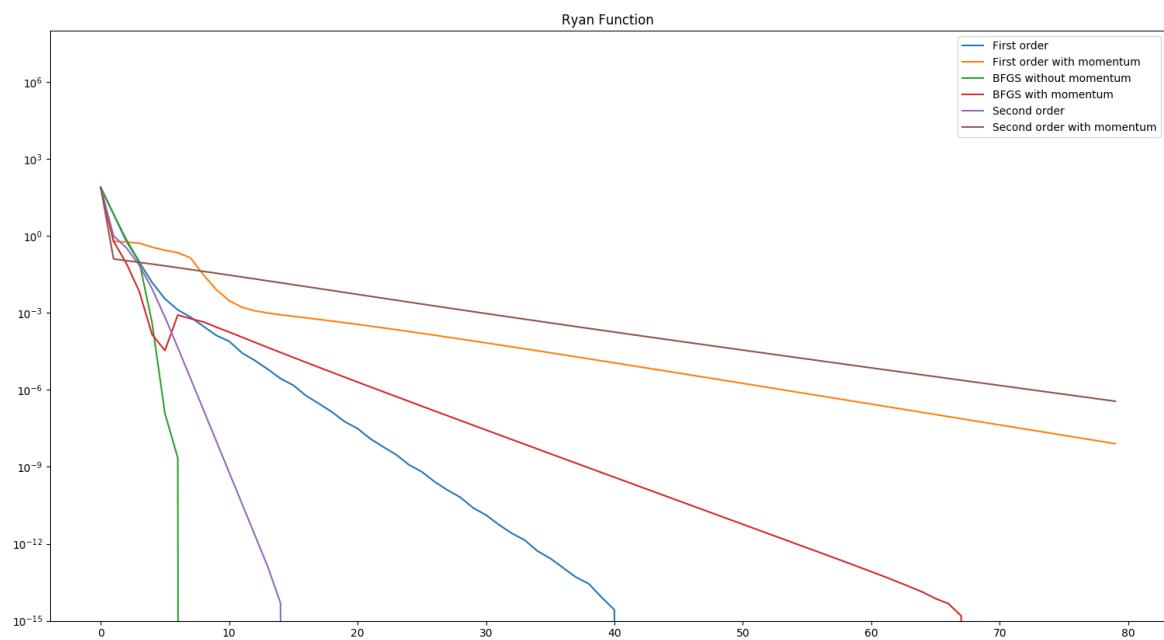


Figure 20: With Backtracking nesterov momentum convergence for Ryan's function

8.2 Non-Convex Functions

8.2.1 Rosenbrock's function

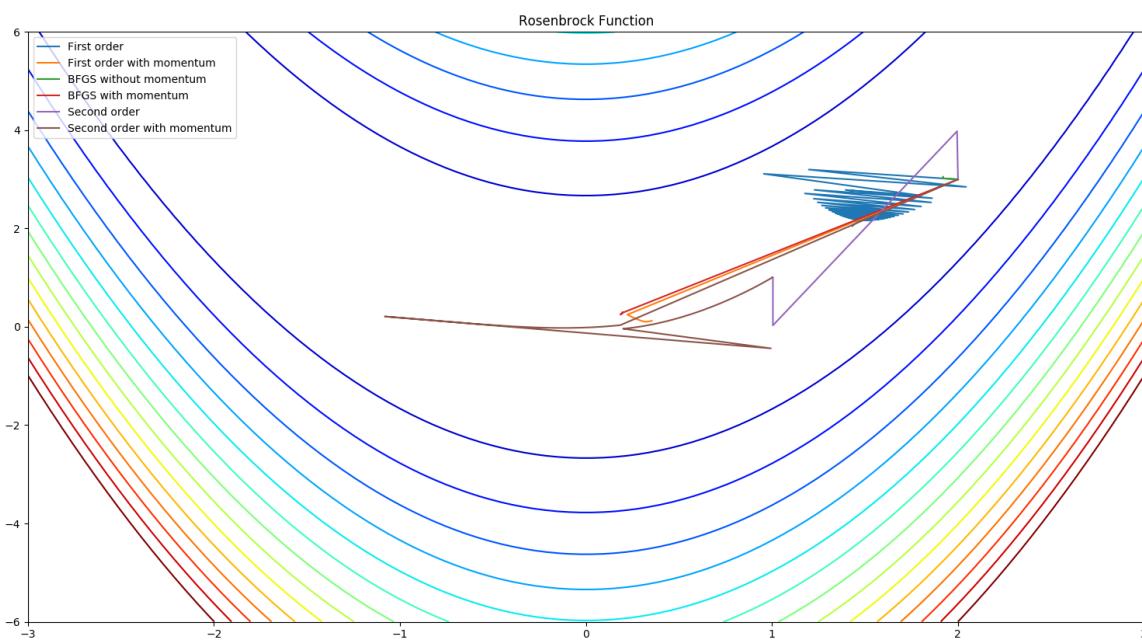


Figure 21: Without Backtracking nesterov momentum for Rosenbrock's function

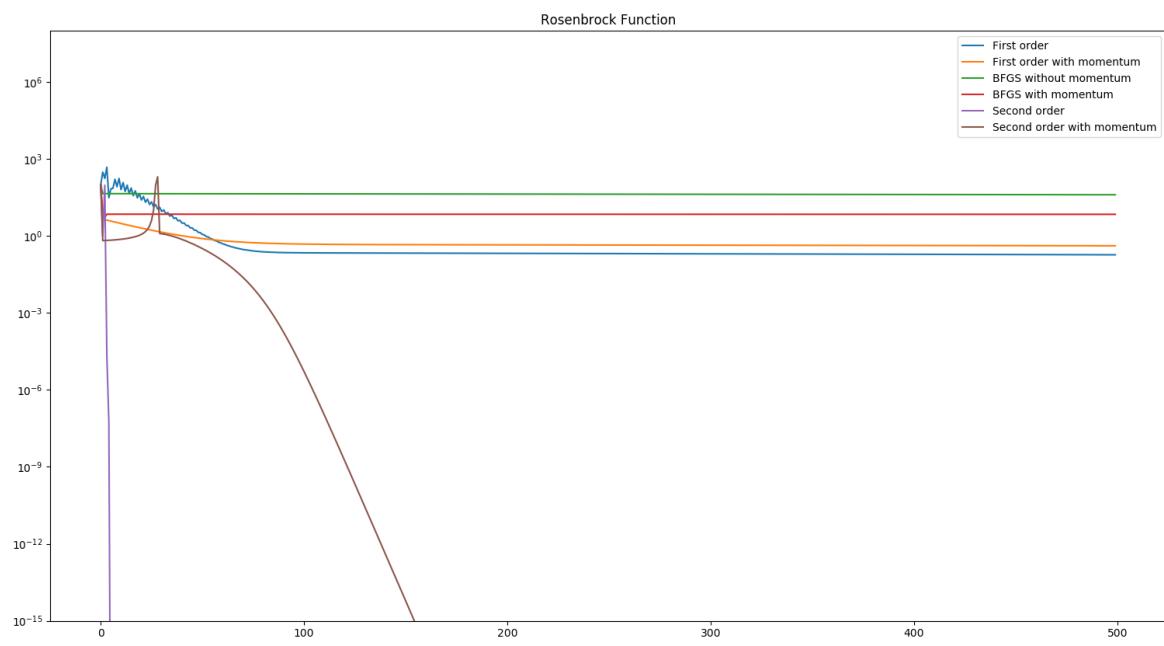


Figure 22: Without Backtracking nesterov momentum convergence for Rosenbrock's function

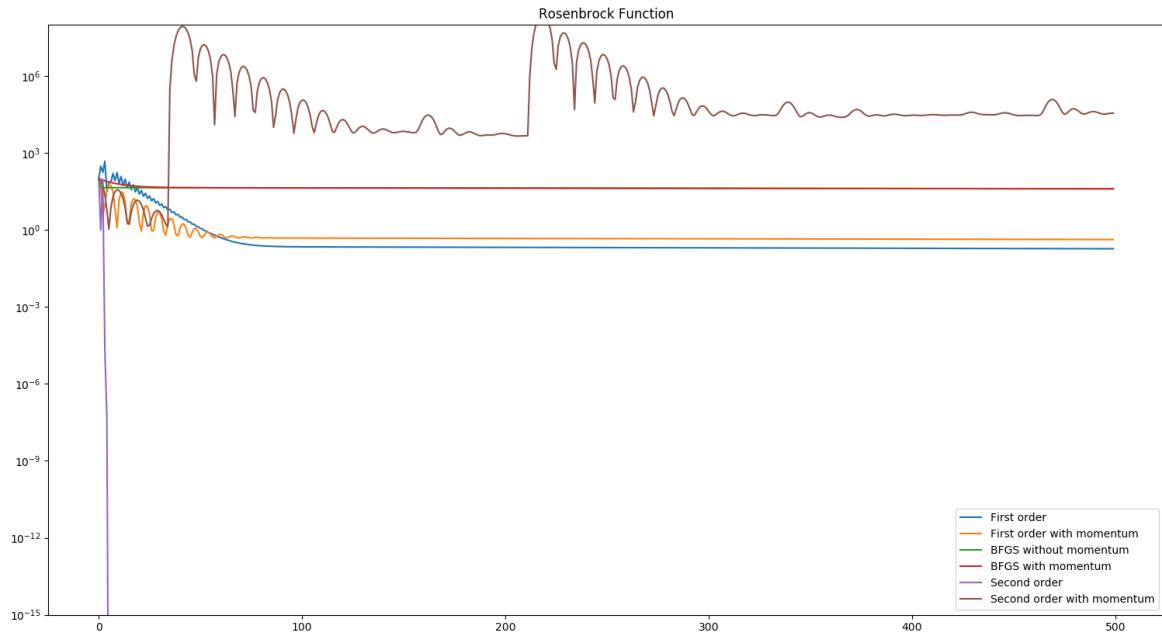


Figure 23: Without Backtracking, classical momentum convergence for Rosenbrock's function

Table 4: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	101	101	101	101	101	101	101	101	101
5	69.829	57.696	3.822	8.157e-24	1.065	0.691	45.096	75.113	100.956
10	64.439	14.953	3.086	8.157e-24	35.252	0.750	100.799	100.916	100.917
20	25.468	1.899	2.048	8.157e-24	13.316	1.166	100.600	100.758	100.758
50	1.148	0.564	0.795	8.157e-24	11667137301325		100.004	100.181	100.181
100	0.222	0.488	0.486	8.157e-24	116263.7955.219e-06		99.019	99.194	99.194
250	0.207	0.464	0.447	8.157e-24	4879335.966.212e-16		96.122	96.289	96.289
500	0.186	0.427	0.417	8.157e-24	35898.283	7.212e-16	91.482	91.638	91.638

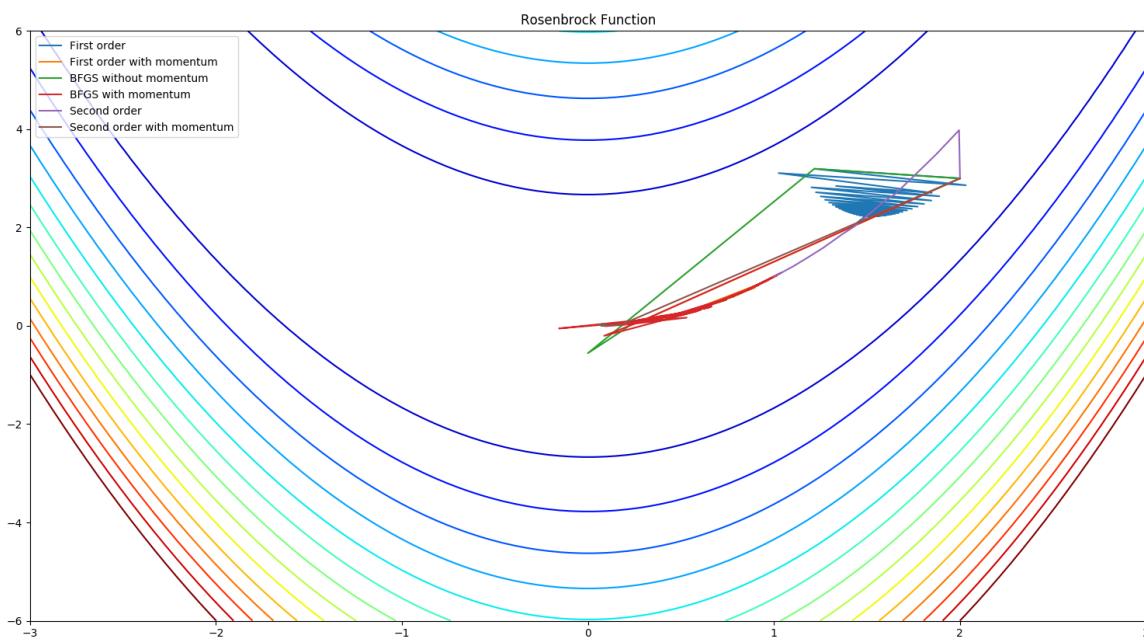


Figure 24: With Backtracking nesterov momentum for Rosenbrock's function

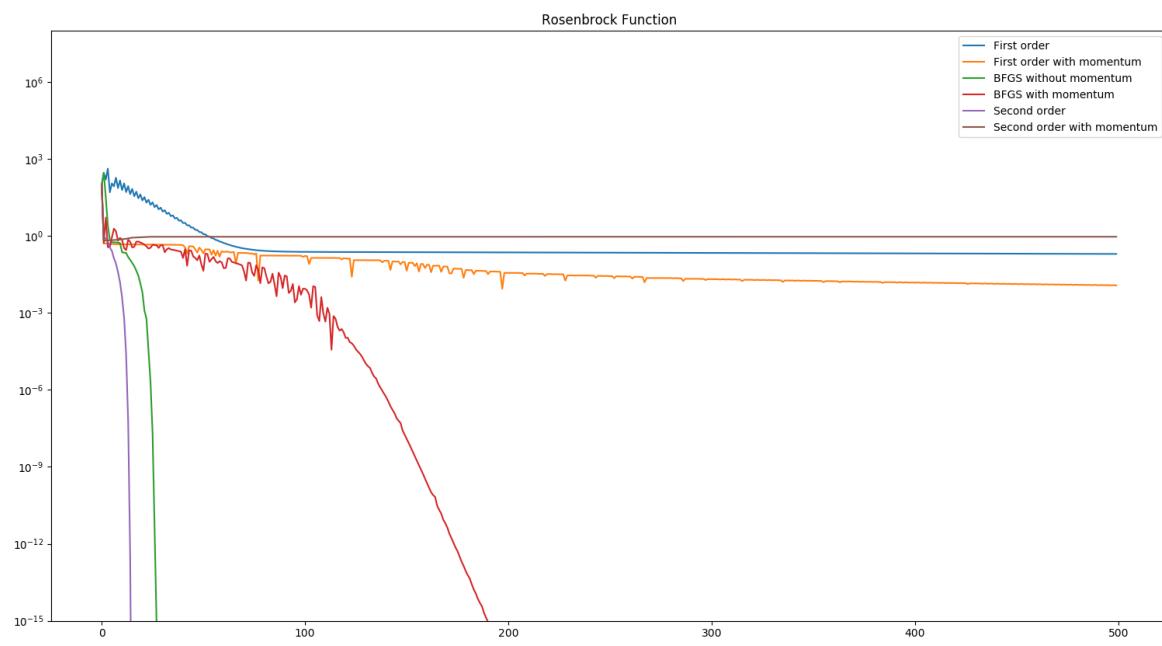


Figure 25: With Backtracking nesterov momentum convergence for Rosenbrock's function

8.2.2 Beale's function

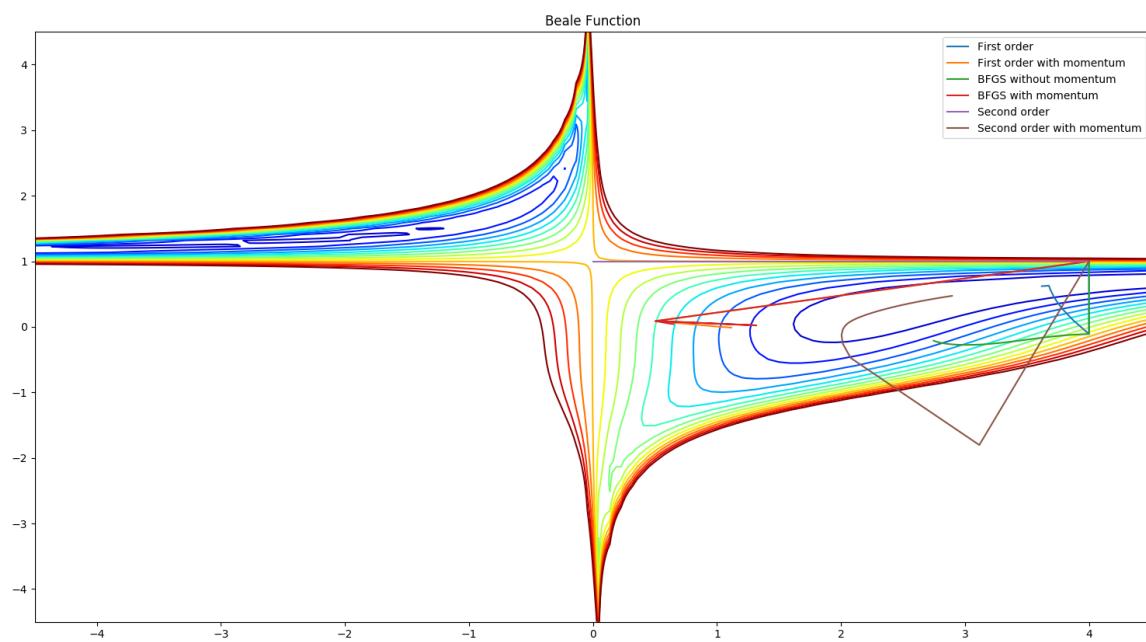


Figure 26: Without Backtracking nesterov momentum for Beale's function

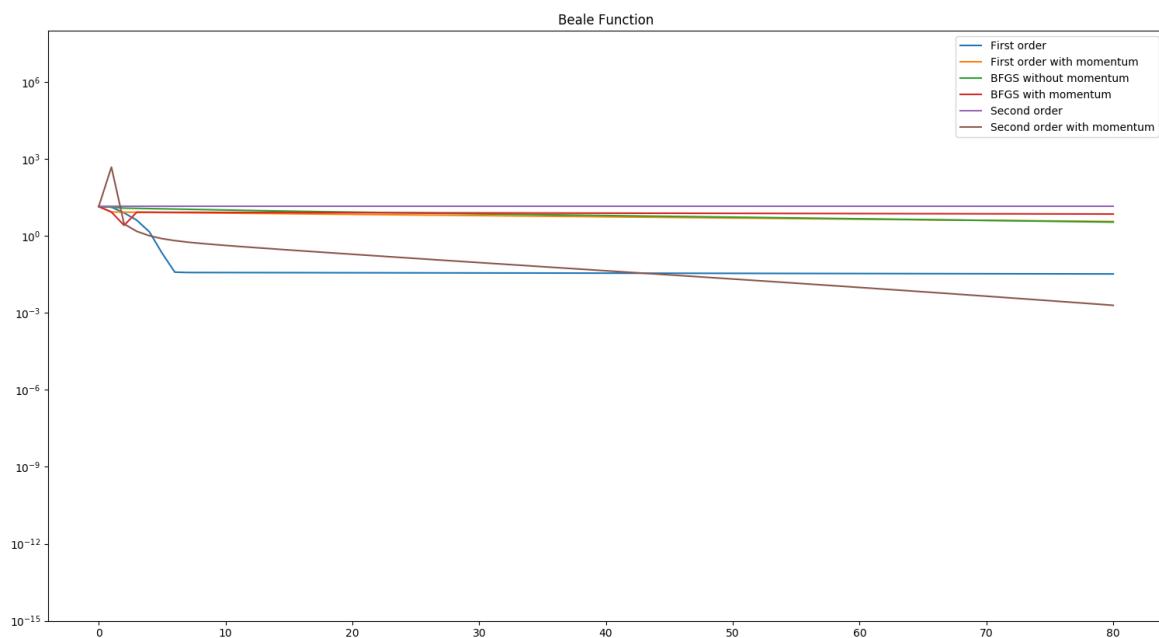


Figure 27: Without Backtracking nesterov momentum convergence for Beale's function

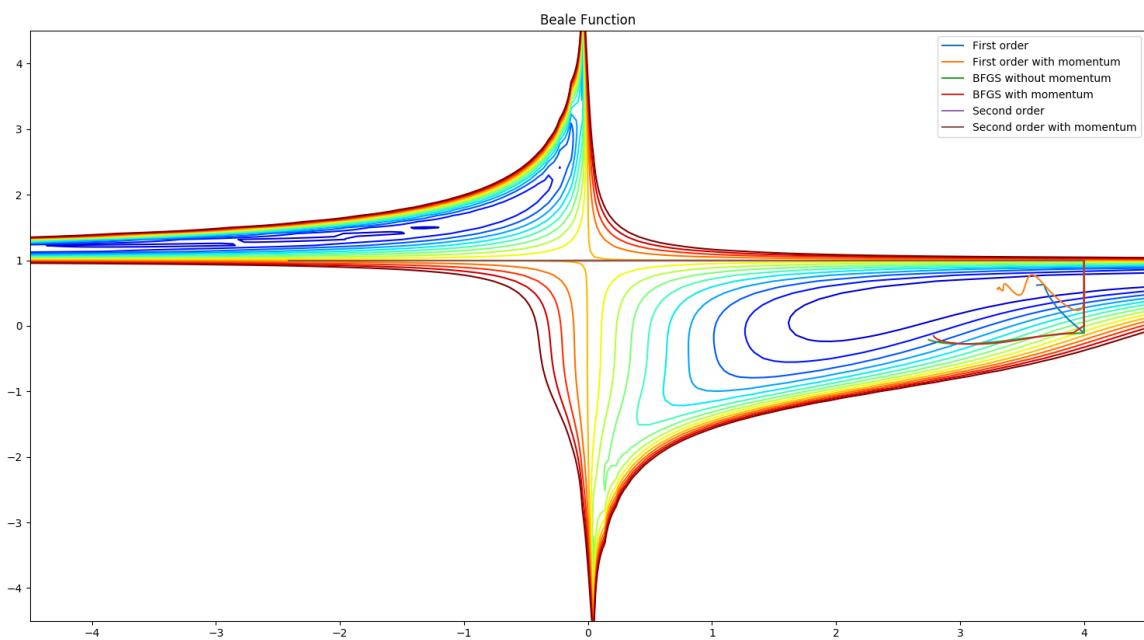


Figure 28: Without Backtracking classical momentum for Beale's function

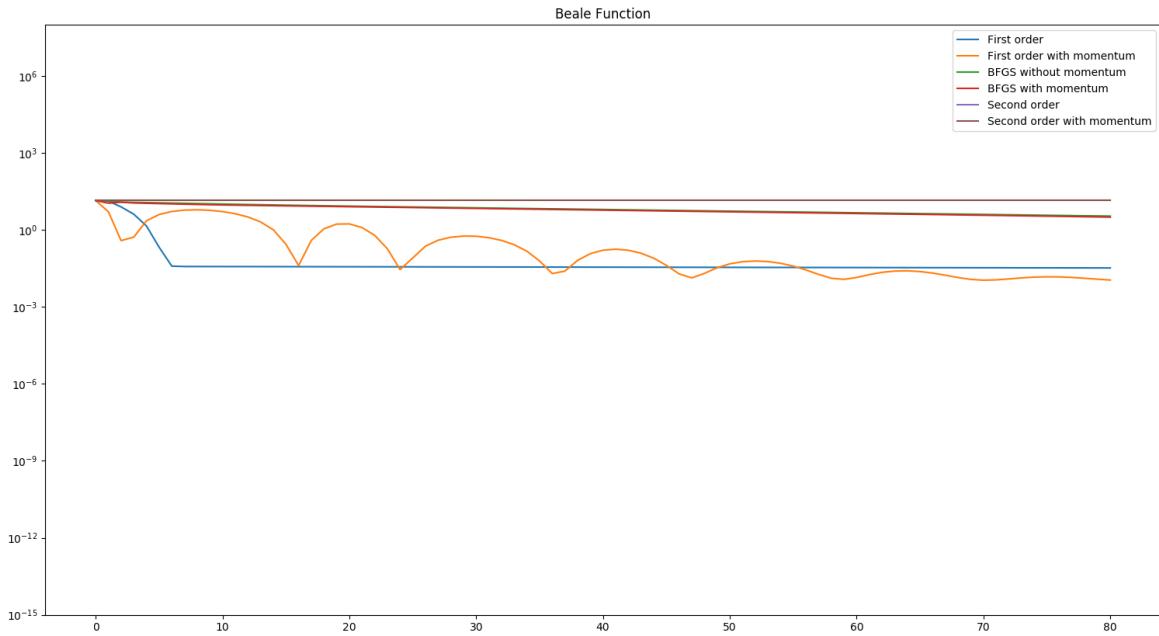


Figure 29: Without Backtracking, classical momentum convergence for Beale's function

Table 5: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	14.203125	14.203125	14.203125	14.203125	14.203125	14.203125	14.203125	14.203125	14.203125
5	0.214	4.050	8.293	14.203	14.203	0.793	11.566	0.988	11.464
10	0.0377	5.262	7.850	14.203	14.203	0.431	9.079	12.489	7.280
20	0.037	1.741	7.0314	14.203	14.203	0.195	2.883	286.229	1.498
50	0.0352	0.0482	5.0662	14.203	14.203	0.021	9.270	0.273	0.219
100	0.0321	0.0101	3.0494	14.203	14.203	0.000358	31641529	66.204	0.4929
250	0.0235	0.00568	1.0456	14.203	14.203	4.345e-10	13.033	0.305	8.958
500	0.012	0.00188	0.353	14.203	14.203	9.454e-16	13.024	0.389	0.0964

8.2.3 Bird's function

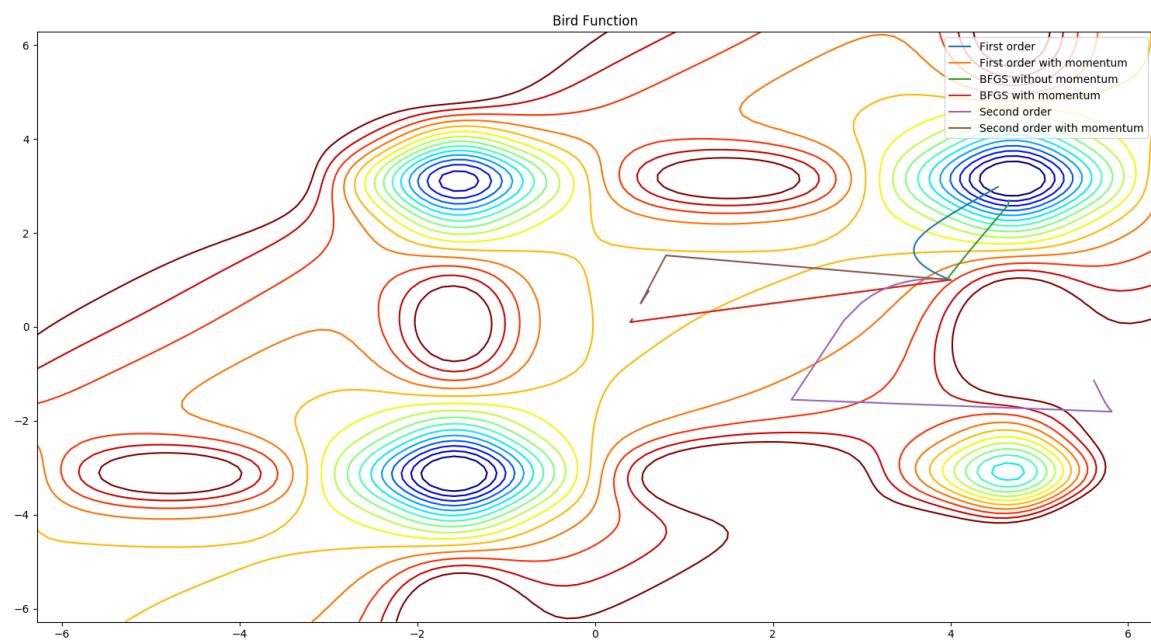


Figure 30: Without Backtracking nesterov momentum for Bird's function

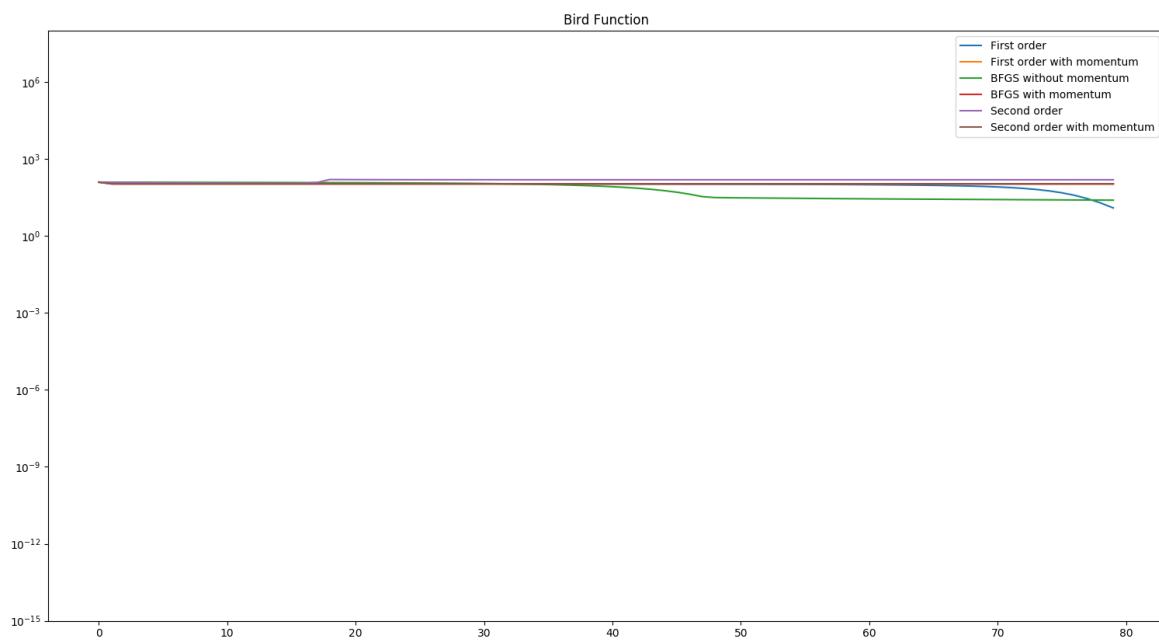


Figure 31: Without Backtracking nesterov momentum convergence for Bird's function

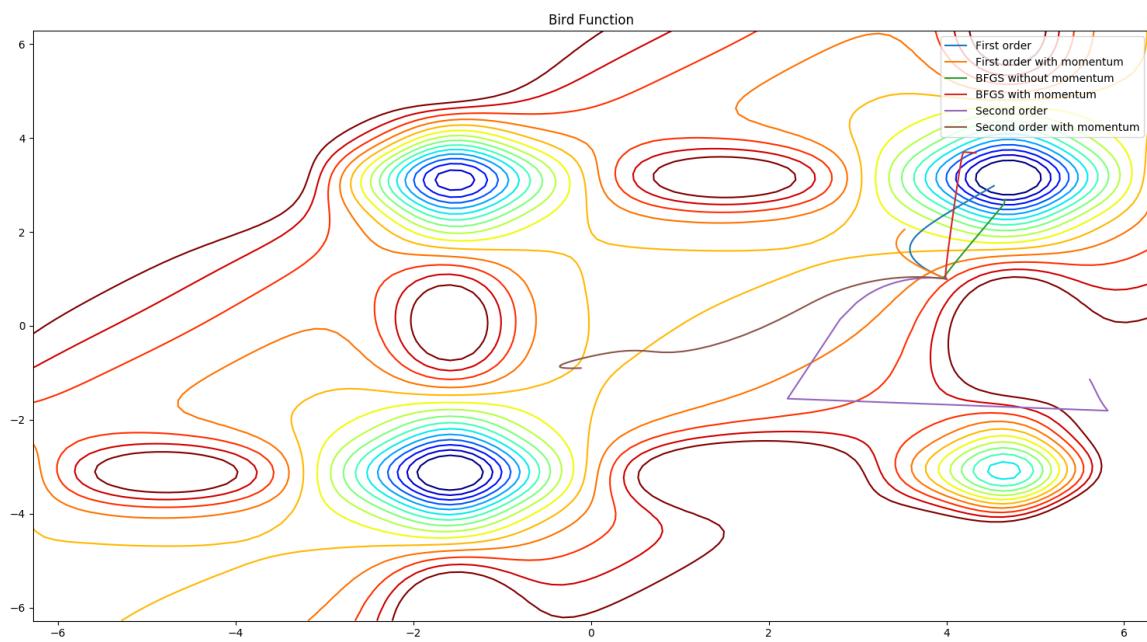


Figure 32: Without Backtracking classical momentum for Bird's function

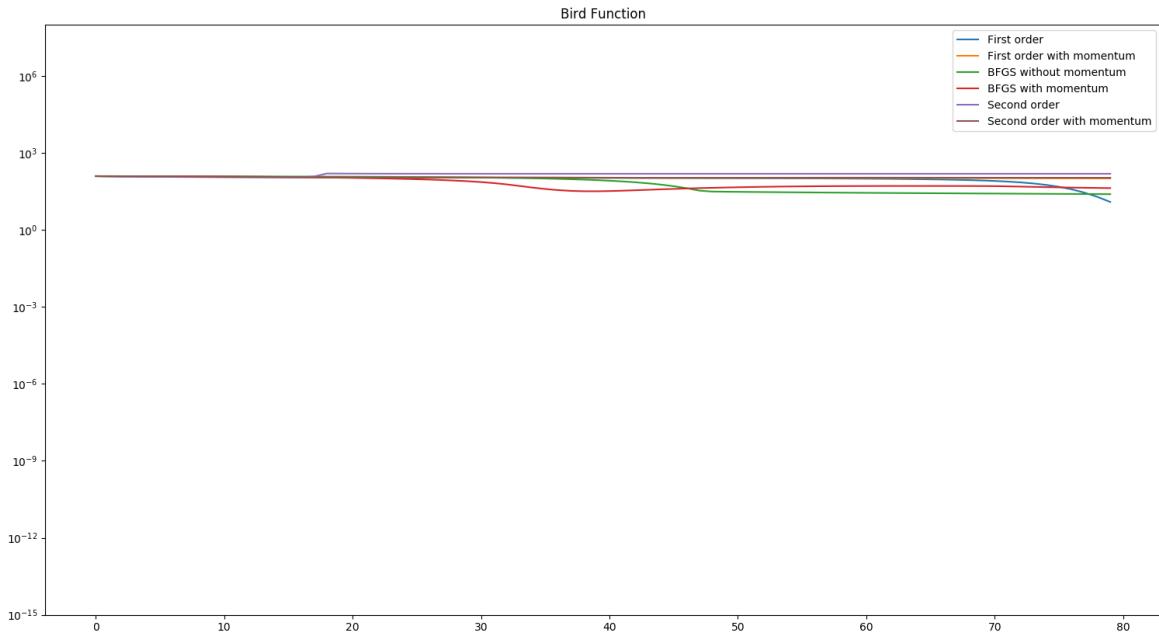


Figure 33: Without Backtracking, classical momentum convergence for Bird's function

Table 6: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	126.66071	14326.66071	14126.66071	14126.66071	14126.66071	14126.66071	14126.66071	14126.66071	14126.66071
5	120.305	124.579	108.687	119.121	123.940	108.401	124.496	125.477	124.873
10	116.747	120.922	108.687	115.838	119.659	108.390	124.927	120.497	120.497
20	112.812	114.799	108.686	159.228	114.050	108.358	121.092	102.870	102.870
50	105.720	108.187	108.685	157.280	109.488	108.301	122.571	53.609	110.140
100	2.592e-05	82.502	108.682	157.277	109.264	108.267	124.468	62.815	62.815
250	2.507e-07	9.538e-06	108.673	157.277	109.257	108.252	109.884	27.930	207060.188
500	2.507e-07	2.507e-07	108.660	157.277	109.257	108.251	14.137	114.811	118.692

8.2.4 Levy's function

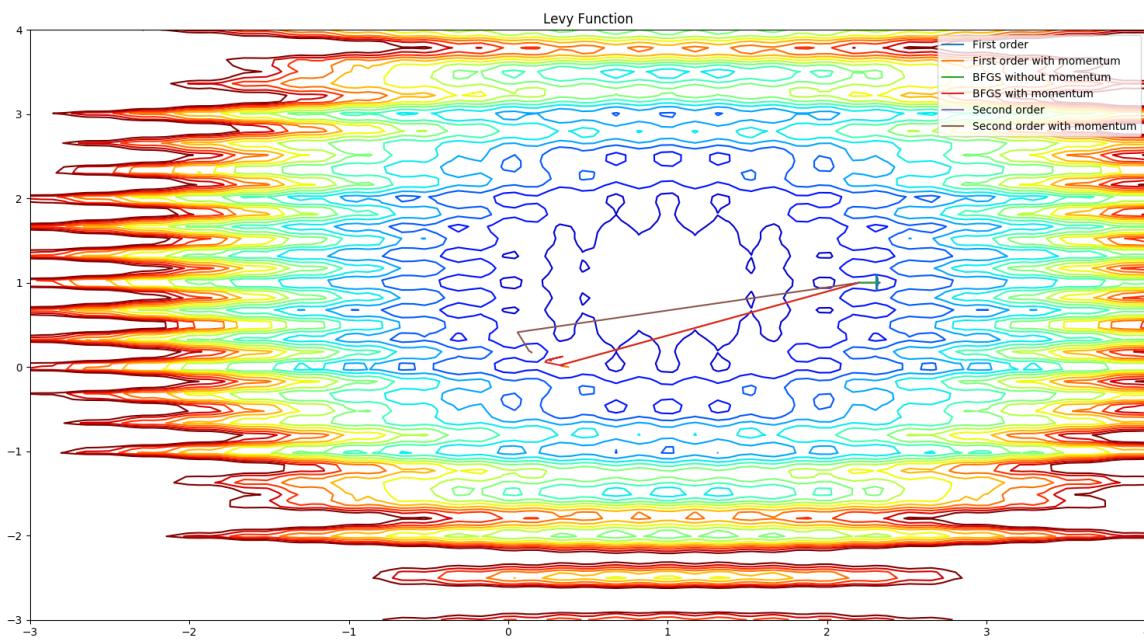


Figure 34: Without Backtracking nesterov momentum for Levy's function

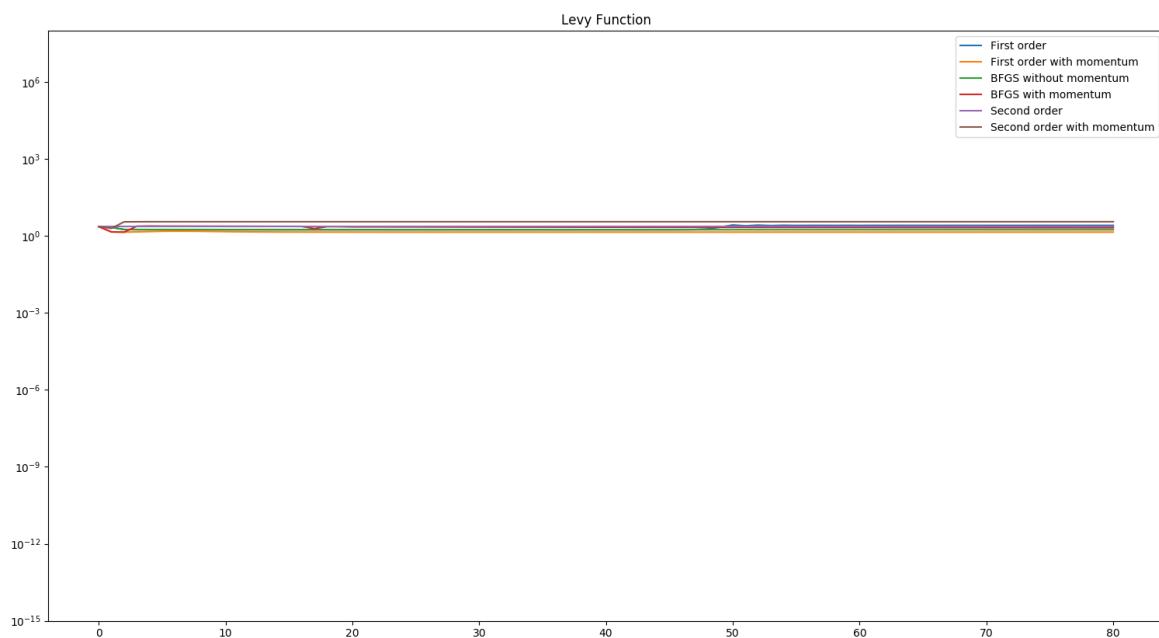


Figure 35: Without Backtracking nesterov momentum convergence for Levy's function

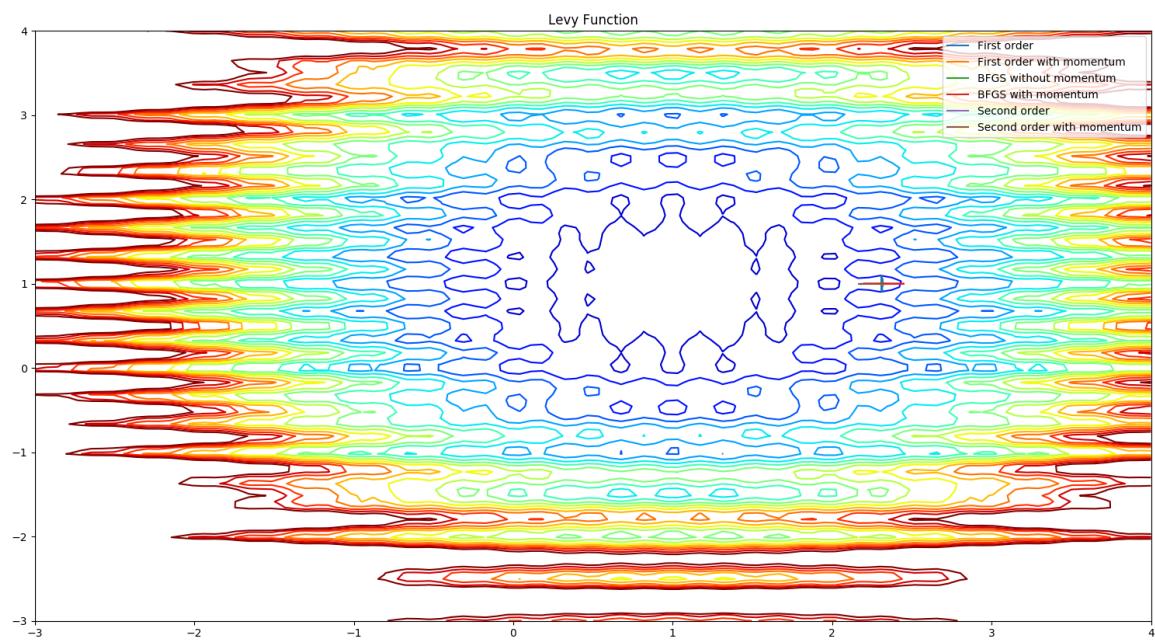


Figure 36: Without Backtracking classical momentum for Levy's function

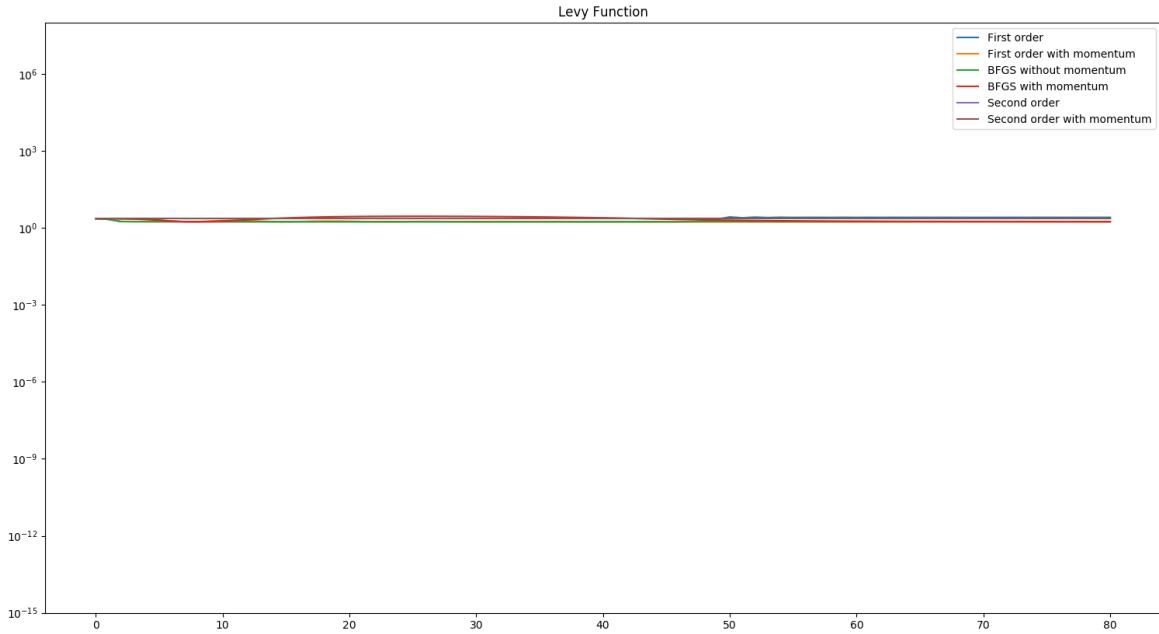


Figure 37: Without Backtracking, classical momentum convergence for Levy's function

Table 7: $|f(x^k) - f(x^*)|$ after k iterations

K	Gradient Descent	Gradient Descent with Momentum	Gradient Descent with Nesterov Momentum	Newton's Method	Newton's Method with Momentum	Newton's Method with Nesterov Momentum	BFGS	BFGS with Momentum	BFGS with Nesterov Momentum
0	2.344	2.344	2.344	2.344	2.3445	2.344	2.344	2.344	2.344
5	1.761	2.0345	1.539	2.376	2.376	3.636	1.796	2.034	2.034
10	1.758	1.972	1.49	2.376	2.364	3.638	2.336	1.879	2.342
20	1.757	1.837	1.428	2.376	2.372	3.638	2.327	2.337	2.337
50	2.711	1.760	1.426	2.376	2.376	3.638	9.947	2.313	2.313
100	2.610	1.757	1.426	2.376	2.376	3.638	2.220	2.245	2.245
250	2.610	1.757	1.426	2.376	2.376	3.638	1.762	2.258	1.771
500	2.610	1.757	1.426	2.376	2.376	3.638	1.757	3.955	1.757

9 Observations and Inferences

We now describe our observations and inferences based on the results of our experiments.

We first analyze the results on using backtracking line search. The results can be observed in figures 6, 8, 14, 20, 25. We observe that with backtracking line search, in every single case, using either momentum or Nesterov momentum performs worse than without any acceleration. This leads us to believe that using momentum might not be a good choice along with a backtracking step.

Proposition 1 Both momentum and Nesterov’s momentum decrease the performance of every method if the step size is chosen using backtracking line search.

Next, we look at the performance of momentum and Nesterov momentum when used with Newton’s method. For momentum, the results can be visualized in figures 4, 12, 18, 23, 29. Similarly, for Nesterov Momentum, the results can be seen in figures 2, 10, 16, 22, 27. In 29 and 27, which correspond to the Beale Function, we observe that Newton’s method with momentum and Nesterov’s momentum performs better than all other methods. However, in all other cases, using acceleration with Newton’s method is always worse than without. This suggests that the existing formulation of momentum is not suitable for Newton’s method.

Proposition 2 Both momentum and Nesterov momentum appear to, in most cases to decrease the performance of Newton’s method if the step size is fixed.

We next analyze the performance of acceleration on BFGS. The results with Nesterov momentum can again be seen in figures 2, 10, 16, 22, 27, and with classical momentum can be seen in 4, 12, 18, 23, 29. We observe differences in behaviour with different functions. In 4, 29, 22, and 27, we observe that on the log-scale plot, the performance of BFGS with and without acceleration is very similar. In 12, 23, and 10, use of acceleration significantly improves the performance of BFGS. However, in 18, 2, and 16, BFGS without acceleration performs better than BFGS with acceleration. We see that even for non-convex functions, BFGS performs comparably well with and without acceleration. Using acceleration for BFGS thus appears to be a promising direction for future work, and needs to be investigated further.

Proposition 3 Momentum and Nesterov momentum perform comparable - in some cases better, in some cases worse - than optimization without using them, when used on BFGS if the step size is fixed.

9.1 Limitations

Our experiments test over a very small set of benchmark functions. It is quite possible that our results and conclusions are biased due to the choice of functions, and that momentum for second order methods performs much better for a different set of functions. However, we believe that this might not be the case, as the functions were chosen with the intent of having as diverse a set of properties among them as possible

10 Conclusion

Since backtracking line search is generally used along with Newton’s method, our experiments suggest that we should avoid using momentum with Newton’s method. There is no clear advantage on using momentum for BFGS. We also observe that momentum performs poorly, even for gradient descent, when used with backtracking line search.

Momentum for first order methods is based on the intuitive idea of damping gradient descent, which allowed the step sizes to be increased. Over the past many decades, strong convergence guarantees have been proven for using momentum. Nesterov’s momentum, proposed in 2013 by Sutskever et al. [2], has performed very well for deep learning. In the case of second order methods, an alternative, more intuitive formulation for acceleration might be required, that can exploit the assumptions made in these methods naturally, as opposed to direct application of the first order formulation.

References

- [1] B.T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1 – 17, 1964. ISSN 0041-5553. doi: [https://doi.org/10.1016/0041-5553\(64\)90137-5](https://doi.org/10.1016/0041-5553(64)90137-5). URL <http://www.sciencedirect.com/science/article/pii/0041555364901375>.
- [2] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML’13, pages III–1139–III–1147. JMLR.org, 2013. URL <http://dl.acm.org/citation.cfm?id=3042817.3043064>.
- [3] William C. Davidon. Variable metric method for minimization. *SIAM Journal on Optimization*, 1(1):1–17, 1991. doi: 10.1137/0801001. URL <https://doi.org/10.1137/0801001>.
- [4] Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965.
- [5] Jorge Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of computation*, 35(151):773–782, 1980.

- [6] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.
- [7] Momin Jamil and Xin-She Yang. A literature survey of benchmark functions for global optimisation problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, 4(2):150–194, 2013.