ADAPTIVE STEP SIZE SELECTION FOR OPTIMIZATION VIA THE SKI RENTAL PROBLEM

Amirali Aghazadeh, Ali Ayremlou, Daniel D. Calderón, Tom Goldstein

Raajen Patel, Divyanshu Vats, Richard G. Baraniuk

Rice University
Electrical and Computer Engineering
6100 Main Street, Houston, TX, 77005

ABSTRACT

Optimization has been used extensively throughout signal processing in applications including sensor networks and sparsity based compressive sensing. One of the key challenges when implementing iterative optimization algorithms is to choose an appropriate step size for fast algorithms. We pose the problem of choosing step sizes as solving a ski rental problem, a popular class of problems from the computer science literature. This results in a novel algorithm for adaptive step size selection that is agnostic to the choice of the optimization algorithm. Our numerical results show the advantages of using adaptivity for step size selection.

Index Terms— ski rental problem, sensor networks, step size using adaptivity, sparsity

1. INTRODUCTION

Consider the following structured optimization problem:

$$x^* = \min_x f(x) + \lambda r(x) \,, \qquad (1)$$

where $f(x)$ is an objective function and $r(x)$ is some regularization term that imposes structure on the solution x^*. The above problem finds applications in many fields. For example, consider a large sensor network where each sensor is a phasor measurement unit (PMU) that monitors electrical waves in an electricity grid. With the current infrastructure, the PMU devices are the only real time measurements available to sensor network operators. Using these PMU measurements, it is of interest to detect power line failures in *real-time* so that operators can take necessary steps to avoid a large blackout in the sensor network. Reference [1] shows how this problem can be posed as an optimization problem of the form (1), where the power line outages are detected using the sparsity pattern of x^*.

Typical algorithms for solving (1) only compute an approximation of the true optimal solution x^* using iterative methods. Most such algorithms require the user to choose a step size parameter. The efficiency of the algorithm depends on this choice in an essential way. However, the user usually has no a-priori knowledge of how to choose this parameter, and thus adaptive schemes must be used. For some specific problems and algorithms there are well known adaptive schemes, while for others little is known. This motivates our problem: *can we design an adaptive step size in a way that is agnostic to the choice of iterative method?*

Our framework will be applicable to most splitting and gradient methods. Most algorithms in these categories require a step size to be chosen. Our adaptive scheme is motivated by the Ski Rental Problem, a class of problems in which there is a choice for a particular task between paying a repeated cost versus paying a one time cost (going skiing each day for an unknown number of days, for example) [2]. For our problem, the task is to achieve a certain level of unknown precision. The one time cost is the computations required for finding a good step-size. The repeated cost is the cost of performing one iteration. Using the break-even solution of the ski rental problem, we propose a simple adaptive scheme for choosing step-sizes when solving a generic optimization problem. The advantage of using our framework is that it leads to the best non deterministic algorithm for choosing step-sizes in a generic optimization algorithm.

2. PRIOR WORK

Several algorithms have been proposed in the literature for choosing step sizes in optimization algorithms. The most accurate method is exact line search, where a separate optimization problem is solved to find the step size [3]. However, solving this optimization problem can be time consuming. For this reason, most line searches rely on inexact methods such as backtracking line search, where the step size is chosen adaptively. However, in backtracking schemes the step size decreases monotonically regardless of the energy function. Also, in many optimization scenario fixing the step size rather than backtracking is more cost effective (see for exam-

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

ple, Armijo-Wolfe Line search [4] [5]). The main feature of our algorithm is that it can be applied to a generic optimization algorithm. This is what differentiates our algorithm from other algorithms in the literature.

3. THE SKI RENTAL PROBLEM

The ski rental problem [2], also known as the rent-or-buy problem [6], is a class of problems where one needs to find an online strategy of either paying repeated costs to perform a task, or paying a one-time cost to perform a task.

Suppose, in a ski rental scenario, that the cost of renting skis is s dollars and buying skis costs C dollars. The ski rental problem here is to decide a day d such that we rent skis for $d-1$ days and then buy skis on the dth day of skiing. If we know that we are going to ski for D days, it is easy to see that choosing $d = \min\{C/s, D\}$ is optimal for minimizing the cost of skiing. However, in practice, D is *not known*. In this case, the following break-even strategy is typically used:

Break-even strategy: Buy skis on day $d = C/s$. (2)

It is known [7] that the above break-even strategy is optimal in the sense that it minimizes the ratio of what we pay by renting for $d-1$ days and buying on the dth day and what we would have paid if the number of days we ski were known in advance.

4. ADAPTIVE STEP SIZE SELECTION

Motivated by the ski rental problem, we now outline our algorithm for choosing the step-size when applying iterative methods to solve the problem in (1). Consider a generic iterative optimization algorithm for solving (1) that depends on a step-size τ_k. Every iteration of the algorithm produces two outputs, (i) the next iterate x_{k+1}, and (ii) a *residual* that measures how far the iterate lies from the true solution. Let C be the cost (in iterations) of updating the adaptive step size. This corresponds to the cost of buying skis in the ski rental problem. For most adaptive schemes, C is *known*. After each iteration, we either

1. Proceed with the current step size, or
2. Perform an adaptive step to obtain an improved step size

Clearly, if the adaptive timestep search is expensive, then the algorithm performance may suffer if we update the step size too frequently. Conversely, if the parameter is not updated frequently enough, runtimes will be long because of slow convergence. This motivates the following problem: *what is the optimal schedule for updating the step size parameter?*

It is clear that finding the optimal schedule for updating the step size is equivalent to finding the number of days to rent skis in the ski rental problem. We wish to apply the break even strategy in (2) to find the optimal update schedule.

The user of our method requires some number of digits of precision, D. This is analogous to the number of days that skis are to be rented. Using the initial step size, each digit of precision costs I iterations. Suppose that the cost of achieving a digit of precision after the adaptive update is only $(1-F)I$, for some $0 < F < 1$. Then the cost of obtaining D digits using the original step size is ID. If the adaptive update is used immediately, the cost of attaining D digits is $ID(1-F)+C$. The break even rule states that we should update the step size when these two costs are even, i.e., when $ID(1-F)+C = ID$. It follows that we should compute $D = \frac{C}{IF}$ digits of precision before we update the steps size. If we multiply this result by I to account for the fact that each digit required I iterations, then it become clear that we should update the step size every C/F iterations.

5. IMPLEMENTATION OF THE ALGORITHM

Algorithm 1 Adaptive Minimization Scheme

```
Initialize:
0 < F < 1, t > 0, γ > 1, C > 0
B ← C/F
x ∈ R^n, r ← 2ε
k ← 0
while ||r|| ≥ ε do
    r_0 ← r
    k ← k + 1
    {x, r} ← Iteration(x, t)
    I = 1/log(r/r_0)
    if k = B then
        {x_1, r_1} ← Iteration(x, t/γ)
        {x_2, r_2} ← Iteration(x, t)
        {x_3, r_3} ← Iteration(x, tγ)
        i ← arg min_{j=1,2,3} ||r_j||
        I_0 ← I
        I ← 1/log(r_i/r)
        F ← 1 − I/I_0
        x ← x_i ; r ← r_i
        k ← 0
        B ← C/F
    end if
end while
return x
```

Our algorithm is outlined in Algorithm 1. We initialize a line search parameter γ, an estimated cost savings F, and a step size t. We then define the optimal/break-even number of iterations to be $B = C/F$. The algorithm proceeds as follows:

- On each iteration, an estimate of the signal x and its residual r is obtained using the chosen iterative method.

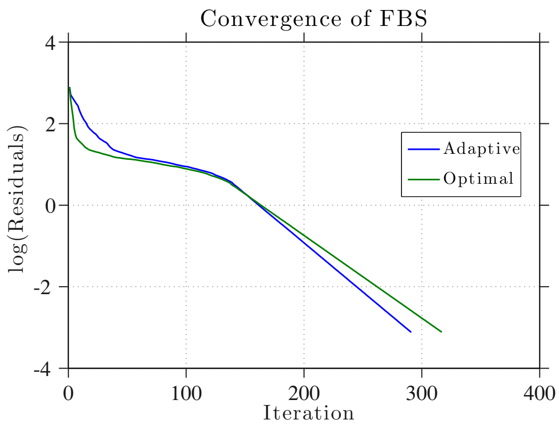

Fig. 1. The adaptive and conventional schemes applied to the LASSO regression problem. Convergence curves show the relative error vs iteration number. Note that the non-adaptive scheme with optimized step size performs very well for the first 10 iterations, but the adaptive scheme quickly becomes more efficient once an effective stepsize is identified.

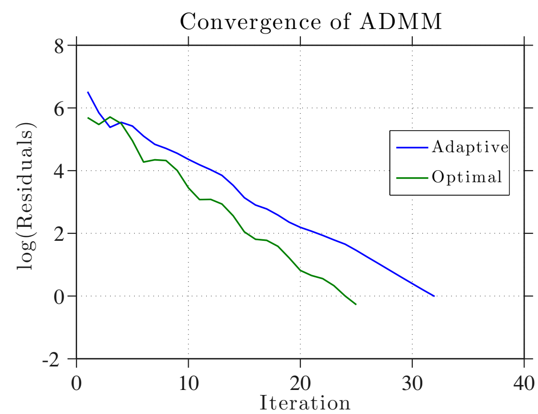

Fig. 2. Results of ADMM applied to the compressive sensing image reconstruction problem. The adaptive scheme attains performance close to that of the conventional scheme with optimized step size parameter.

We refer to this iterative method as $Iteration(\cdot, \cdot)$ in Algorithm 1.

- If the number of iterations k is equal to B, we "buy our skis" by searching for a new time step. This is done by performing an iteration with step size $t/gamma$, t, and γt, and then choosing the step size that yields the minimal residual. Note that this corresponds to $C = 3$, i.e., the cost of updating the step size.

- We then update the value of F to be ones minus the ratio of the estimated convergence rate of the new step size to the convergence rate of the original step size. Using this constant F, we update the break even number of iterations to $B = C/F$, and proceed as before.

Remark: While the user has the freedom to choose the parameters, we recommend the following: $F = 0.1, \gamma = 2$.

6. RESULTS

In this section we will show numerical simulations to compare adaptive and non-adaptive methods. We will apply our framework to two different iterative methods: Forward-backward splitting (FBS) [8], and the Alternating Direction Method of Multipliers (ADMM) [9]. Both of these algorithms are very general, and are commonly applied to minimizations where sparsity of the results is desired.

The first test problem is a sparse regression problem involving large random matrices and the LASSO regression [10]. We wish to minimize the energy

$$\lambda||x|| + \frac{1}{2}\|Ax - b\|^2$$

where A is a random gaussian matrix with dimensions 100×1000. The true solution x_0 is a sparse random vector containing 30 non-zero elements. The measurement data is $b = Ax_0 + \eta$, where η is a random Gaussian vector with standard deviation 10^{-3}. Recovery was performed with $\lambda = 10$.

For comparison, we apply the FBS method using both the adaptive and non-adaptive versions. The non-adaptive algorithm was applied using an optimal step size, which was determined using a script which tuned the stepsize to achieve the lowest possible iteration count. In practice, the user does not know this optimal timestep a-priori (which is why a adaptive scheme is desired), and so these curves represent the best-case performance attainable with adaptivity. In many cases, the adaptive scheme is capable of achieving performance comparable to this optimal rate, without any a-priori knowledge of the ideal step size. The adaptive method was applied using a starting step size of $t = 0.001$, which is approximately an order of magnitude smaller than the optimal value.

The FBS method was applied to the problem, and iterations were terminated when the residual dropped below a specified tolerance. Results are shown in Table 1 and Figure 1. Note that the adaptive scheme performs well over a wide range of tolerance parameters, ranging from 2 to 7 digits of precision. In all but one experiment, the adaptive scheme out-performs the non-adaptive scheme using an optimal parameter. This is possible because the behavior of the algorithm changes as the active set of the solution evolves. By updating the stepsize as the iterations progress, the adaptive

Table 1. Comparison of the non-adaptive algorithm with our adaptive method

Tolerance	Optimal t	Number of iteration for non–adaptive	Number of iteration for adaptive
0.01	0.00158321	84	69
0.001	0.0014636419	155	159
0.0001	0.001350814727174	217	200
0.00001	0.001351961110898	267	242
0.000001	0.00135200208679	316	284
0.0000001	0.00141270336336	349	325

Fig. 3. The 128 × 128 Shepp-Logan phantom used in the ADMM image processing experiment. The reconstructed image was visually indistinguishable from the input image, and so only one image is shown.

scheme is able to outperform the non-adaptive, constant-step-size method.

The second problem we consider is a compressive sensing problem in which we recover an image from a subset of its Fourier modes. In this case, we wish to recover a 2-dimensional array of pixels, x. The problem can be solved by minimizing the following energy:

$$|\nabla x| + \frac{\mu}{2}\|RFx - b\|^2$$

where R is a diagonal matrix of 1's and 0's, F is a Fourier transform matrix, and b is a vector of observed Fourier modes. The quantity $|\nabla x|$ represents the total-variation semi-norm of x, which enforces sparsity of the solution. Only 25% of the Fourier modes were observed. The test image was a digital Shepp-Logan phantom, as depicted in Figure 3.

The problem is solved using the ADMM scheme with $\mu = 500$, and the algorithm was terminated when 1% relative error was reached. The optimal step size for this problem was 0.029, and the adaptive scheme was started with stepsize of 0.01. Sample convergence curves for this problem are shown in Figure 2. In this case, both adaptive and non-adaptive methods exhibited similar convergence, with the non-adaptive scheme (using an optimized step size parameter) slightly outperforming the adaptive scheme.

7. CONCLUSION

We showed how the problem of choosing step size adaptively in optimization problems can be mapped to the ski renting problem from computer science. Our numerical results show that adaptivity in choosing step sizes results in superior algorithms. Moreover, our proposed algorithm for choosing step sizes using adaptivity is agnostic to the optimization problem. We showed this in our numerical simulations where we applied our methods for two different optimization algorithms for solving the same optimization problem. This shows the possible application of our algorithm in various problems including sensor networks, sparsity based signal processing and adaptivity based sensing.

8. REFERENCES

[1] H. Zhu and G. Giannakis, "Sparse overcomplete representations for efficient identification of power line outages," *Power Systems, IEEE Transactions on*, vol. 27, no. 4, pp. 2215 –2224, nov. 2012.

[2] R. Motwani and P. Raghavan, *Randomized algorithms*. Cambridge university press, 1995.

[3] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[4] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives." *Pacific Journal of mathematics*, vol. 16, no. 1, pp. 1–3, 1966.

[5] P. Wolfe, "Convergence conditions for ascent methods," *SIAM review*, vol. 11, no. 2, pp. 226–235, 1969.

[6] R. El-Yaniv, R. Kaniel, and N. Linial, "Competitive optimal on-line leasing," *Algorithmica*, vol. 25, no. 1, pp. 116–140, 1999.

[7] A. Karlin, M. Manasse, L. Rudolph, and D. Sleator, "Competitive snoopy caching," *Algorithmica*, vol. 3, no. 1, pp. 79–119, 1988.

[8] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[10] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.