

**Augmented Lagrangian methods** are a certain class of [algorithms](#) for solving [constrained optimization](#) problems. They have similarities to [penalty methods](#) in that they replace a constrained optimization problem by a series of unconstrained problems and add a penalty term to the [objective](#); the difference is that the augmented Lagrangian method adds yet another term, designed to mimic a [Lagrange multiplier](#). The augmented Lagrangian is not the same as the [method of Lagrange multipliers](#).

Viewed differently, the unconstrained objective is the [Lagrangian](#) of the constrained problem, with an additional penalty term (the **augmentation**).

The method was originally known as the **method of multipliers**, and was studied much in the 1970 and 1980s as a good alternative to penalty methods. It was first discussed by [Magnus Hestenes](#) in 1969<sup>[1]</sup> and by [Powell](#) in 1969.<sup>[2]</sup> The method was studied by [R. Tyrrell Rockafellar](#) in relation to [Fenchel duality](#), particularly in relation to proximal-point methods, [Moreau–Yosida regularization](#), and [maximal monotone operators](#): These methods were used in [structural optimization](#). The method was also studied by [Dimitri Bertsekas](#), notably in his 1982 book,<sup>[3]</sup> together with extensions involving nonquadratic regularization functions, such as [entropic regularization](#), which gives rise to the "exponential method of multipliers," a method that handles inequality constraints with a twice differentiable augmented Lagrangian function.

Since the 1970s, [sequential quadratic programming](#) (SQP) and [interior point methods](#) (IPM) have had increasing attention, in part because they more easily use [sparse matrix subroutines](#) from [numerical software libraries](#), and in part because IPMs have proven complexity results via the theory of [self-concordant functions](#). The augmented Lagrangian method was rejuvenated by the optimization systems LANCELOT and AMPL, which allowed sparse matrix techniques to be used on seemingly dense but "partially separable" problems. The method is still useful for some problems.<sup>[4]</sup> Around 2007, there was a resurgence of augmented Lagrangian methods in fields such as [total-variation denoising](#) and [compressed sensing](#). In particular, a variant of the standard augmented Lagrangian method that uses partial updates (similar to the [Gauss-Seidel method](#) for solving linear equations) known as the [alternating direction method of multipliers](#) or **ADMM** gained some attention.

## General method

Let us say we are solving the following constrained problem:

$$\min f(\mathbf{x})$$

subject to

$$c_i(\mathbf{x}) = 0 \quad \forall i \in I.$$

This problem can be solved as a series of unconstrained minimization problems. For reference, we first list the [penalty method](#) approach:

$$\min \Phi_k(\mathbf{x}) = f(\mathbf{x}) + \mu_k \sum_{i \in I} c_i(\mathbf{x})^2$$

The penalty method solves this problem, then at the next iteration it re-solves the problem using a larger value of  $\mu_k$  (and using the old solution as the initial guess or "warm-start").

The augmented Lagrangian method uses the following unconstrained objective:

$$\min \Phi_k(\mathbf{x}) = f(\mathbf{x}) + \frac{\mu_k}{2} \sum_{i \in I} c_i(\mathbf{x})^2 - \sum_{i \in I} \lambda_i c_i(\mathbf{x})$$

and after each iteration, in addition to updating  $\mu_k$ , the variable  $\lambda$  is also updated according to the rule

$$\lambda_i \leftarrow \lambda_i - \mu_k c_i(\mathbf{x}_k)$$

where  $\mathbf{x}_k$  is the solution to the unconstrained problem at the  $k$ th step, i.e.  $\mathbf{x}_k = \operatorname{argmin} \Phi_k(\mathbf{x})$

The variable  $\lambda$  is an estimate of the [Lagrange multiplier](#), and the accuracy of this estimate improves at every step. The major advantage of the method is that unlike the [penalty method](#), it is not necessary to take  $\mu \rightarrow \infty$  in order to solve the original constrained problem. Instead, because of the presence of the Lagrange multiplier term,  $\mu$  can stay much smaller, thus avoiding ill-conditioning.<sup>[4]</sup>

The method can be extended to handle inequality constraints. For a discussion of practical improvements, see.<sup>[4]</sup>

## Alternating direction method of multipliers

The alternating direction method of multipliers (ADMM) is a variant of the augmented Lagrangian scheme that uses partial updates for the dual variables. This method is often applied to solve problems such as

$$\min_x f(x) + g(x).$$

This is equivalent to the constrained problem

$$\min_{x,y} f(x) + g(y), \quad \text{subject to} \quad x = y.$$

Though this change may seem trivial, the problem can now be attacked using methods of constrained optimization (in particular, the augmented Lagrangian method), and the objective

function is separable in  $x$  and  $y$ . The dual update requires solving a proximity function in  $x$  and  $y$  at the same time; the ADMM technique allows this problem to be solved approximately by first solving for  $x$  with  $y$  fixed, and then solving for  $y$  with  $x$  fixed. Rather than iterate until convergence (like the [Jacobi method](#)), the algorithm proceeds directly to updating the dual variable and then repeating the process. This is not equivalent to the exact minimization, but surprisingly, it can still be shown that this method converges to the right answer (under some assumptions). Because of this approximation, the algorithm is distinct from the pure augmented Lagrangian method.

The ADMM can be viewed as an application of the [Douglas-Rachford splitting algorithm](#), and the Douglas-Rachford algorithm is in turn an instance of the [Proximal point algorithm](#); details can be found here.<sup>[5]</sup> There are several modern software packages that solve [Basis pursuit](#) and variants and use the ADMM; such packages include [YALL1](#) (2009), [SpaRSA](#) (2009) and [SALSA](#) (2009). There are also packages that use the ADMM to solve more general problems, some of which can exploit multiple computing cores [SNAPVX](#) (2015), [parADMM](#) (2016).

## Stochastic optimization

Stochastic optimization considers the problem of minimizing a loss function with access to noisy samples of the (gradient of the) function. The goal is to have an estimate of the optimal parameter (minimizer) per new sample. ADMM is originally a batch method. However, with some modifications it can also be used for stochastic optimization. Since in stochastic setting we only have access to noisy samples of gradient, we use an inexact approximation of the Lagrangian as

$$\hat{\mathcal{L}}_{\rho,k} = f_1(x_k) + \langle \nabla f(x_k, \zeta_{k+1}), x \rangle + g(y) - z^T (Ax + By - c) + \frac{\rho}{2} \|Ax + By - c\|^2 + \frac{\|x - x_k\|^2}{2\eta_{k+1}},$$

where  $\eta_{k+1}$  is a time-varying step size.<sup>[6]</sup>

The alternating direction method of multipliers (ADMM) is a popular method for online and distributed optimization on a large scale,<sup>[7]</sup> and is employed in many applications, e.g.<sup>[8][9][10]</sup>

ADMM is often applied to solve regularized problems, where the function optimization and regularization can be carried out locally, and then coordinated globally via constraints. Regularized optimization problems are especially relevant in the high dimensional regime since regularization is a natural mechanism to overcome ill-posedness and to encourage parsimony in the optimal solution, e.g., sparsity and low rank. Due to the efficiency of ADMM in solving regularized problems, it has a good potential for stochastic optimization in high dimensions. However, conventional stochastic ADMM methods suffer from curse of dimensionality. Their convergence rate is proportional to square of the dimension and in practice they scale poorly. See figure [REASON vs Stochastic ADMM](#)

Recently, a general framework has been proposed for stochastic optimization in high-dimensions that solves this bottleneck by adding simple and cheap modifications to ADMM.,<sup>[11][12]</sup> The method is called **REASON** (Regularized Epoch-based Admm for Stochastic Optimization in high-dimension). The modifications are in terms of added projection which goes a long way and results in logarithmic dimension dependency. REASON can be performed on any regularized optimization with any number of regularizers. The specific cases of sparse optimization framework and noisy decomposition framework are discussed further. In both cases, REASON obtains minimax optimal convergence rate. REASON provides the first online guarantees for noisy matrix decomposition. Experiment results show that in aforementioned cases, REASON outperforms state-of-the-art.

## Alternative approaches

---

- [Sequential quadratic programming](#)
- [Sequential linear programming](#)
- [Sequential linear-quadratic programming](#)

## Software

---

Some well-known software packages that use the augmented Lagrangian method are open source

- [Accord.NET](#) (C# implementation of augmented Lagrangian optimizer)
- [ALGLIB](#) (C# and C++ implementations of preconditioned augmented Lagrangian solver)
- [PENNON](#) (GPL 3, commercial license available)

and non-free/commercial

- [LANCELOT](#) (free "internal use" license, paid commercial options)
- [MINOS](#) (also uses an augmented Lagrangian method for some types of problems).

The code for Apache 2.0 licensed **REASON** is available online.<sup>[13]</sup>

## See also

---

- [Penalty method](#)
- [Interior point method](#)
- [Barrier function](#)
- [Lagrange multiplier](#)

## References

---

1. [M.R. Hestenes](#), "Multiplier and gradient methods", *Journal of Optimization Theory and Applications*, 4, 1969, pp. 303–320
2. M.J.D. Powell, "A method for nonlinear constraints in minimization problems", in *Optimization* ed. by R. Fletcher, Academic Press, New York, NY, 1969, pp. 283–298.
3. Dimitri P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, Athena Scientific, 1996 (first published 1982)
4. [Nocedal & Wright \(2006\)](#), chapter 17
5. Eckstein, J.; Bertsekas, D. P. (1992). "On the Douglas—Rachford splitting method and the proximal point algorithm for maximal monotone operators". *Mathematical Programming*. **55**: 293. doi:[10.1007/BF01581204](#) .
6. Ouyang, H.; He, N.; Tran, L. & Gray, A. G (2013). "Stochastic alternating direction method of multipliers". *Proceedings of the 30th International Conference on Machine Learning*: 80–88.
7. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B. & Eckstein, J. (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning*. **3** (1): 1–122. doi:[10.1561/22000000016](#) .
8. Wahlberg, B.; Boyd, S.; Annergren, M.; Wang, Y. (2012). "An ADMM algorithm for a class of total variation regularized estimation problems". [arXiv:1203.1828](#) .
9. Esser, E.; Zhang, X.; Chan, T. (2010). "A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science". *SIAM Journal on Imaging Sciences*. **3** (4).
10. Mota, J. FC; Xavier, J. MF; Aguiar, P. MQ; Puschel, M. (2012). "Distributed ADMM for model predictive control and congestion control". *Decision and Control (CDC), 2012 IEEE 51st Annual Conference o*.
11. Sedghi, Hanie; Anandkumar, Anima; Jonckheere, Edmond (2014). "Multi-step stochastic ADMM in high dimensions: Applications to sparse optimization and matrix decomposition". *Advances in Neural Information Processing Systems*: 2771–2779.
12. Sedghi, Hanie; Anandkumar, Anima; Jonckheere, Edmond (2014). "Multi-Step Stochastic ADMM in High Dimensions: Applications to Sparse Optimization and Noisy Matrix Decomposition". [arXiv:1402.5131](#) .
13. ["REASON code"](#) .

## Bibliography

- Bertsekas, Dimitri P. (1999), *Nonlinear Programming* (2nd ed.), Belmont, Mass: **Athena Scientific**, ISBN [1-886529-00-0](#)

- Nocedal, Jorge; Wright, Stephen J. (2006), *Numerical Optimization* (2nd ed.), Berlin, New York: Springer-Verlag, ISBN 978-0-387-30303-1

---

**Last edited 4 months ago** by Toomais

---