Interior-Point Methods II and an Introduction to PDHG

How do we compute points near the central path—and how do we know when to stop?

Part	Question
Interior-point update	Which linear equations determine the next step?
Accuracy	How do primal and dual feasibility give an objective bound?
PDHG introduction	Can we update primal and dual variables without a Newton solve?

> We will calculate one complete interior-point step and one complete PDHG step on the same LP.

Recall the LP and all four nonnegative variables

Same LP: minimize $f = -2u - v$, subject to $u + v \leq 4$, $u \leq 3$, $u, v \geq 0$.

Introduce constraint slacks $s_1 = 4 - u - v$ and $s_2 = 3 - u$. Thus $x = (u, v, s_1, s_2)^\top \geq 0$.

$$x = (u, v, s_1, s_2)^\top, \quad A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 4 \\ 3 \end{pmatrix}, \quad c = \begin{pmatrix} -2 \\ -1 \\ 0 \\ 0 \end{pmatrix}.$$

$$Ax = b \quad \Longleftrightarrow \quad u + v + s_1 = 4, \qquad u + s_2 = 3.$$

The optimum is $(u, v) = (3, 1)$, with $x^* = (3, 1, 0, 0)^\top$ and $f^* = -7$.

Keep the primal and dual conditions separate

Same LP: minimize $f = -2u - v$, subject to $u + v \leq 4$, $u \leq 3$, $u, v \geq 0$.

Primal	Dual
$\min c^\top x$	$\max b^\top p = 4p_1 + 3p_2$
$Ax = b,\ x \geq 0$	$A^\top p \leq c,\ p$ unrestricted

$$r = c - A^\top p = \begin{pmatrix} -2 - p_1 - p_2 \\ -1 - p_1 \\ -p_1 \\ -p_2 \end{pmatrix} \geq 0.$$

The vector r contains the four **dual slacks**. It is different from the primal constraint slacks s_1, s_2, which are components of x.

The central-path equations are our target

For a fixed barrier parameter $\mu > 0$, the central-path point satisfies:

$$Ax = b, \qquad A^\top p + r = c, \qquad x > 0,\ r > 0, \qquad x_j r_j = \mu.$$

Here p is unrestricted; strict dual feasibility means $r > 0$, not $p > 0$.

> The equations $x_j r_j = \mu$ replace the LP optimality equations $x_j r_j = 0$. As the positive target μ decreases, these products approach zero.

A numerical iterate usually does not satisfy every product equation exactly. Newton's method computes a correction.

Check an interior starting point before taking a step

Same LP: minimize $f = -2u - v$, subject to $u + v \le 4$, $u \le 3$, $u, v \ge 0$.

Order x = (u, v, s_1, s_2). Start: $x = (1, 1, 2, 2)^\top$, $p = (-3, -2)^\top$, $r = (3, 2, 3, 2)^\top$. Target $\mu = 1$.

$$Ax = \begin{pmatrix} 1+1+2 \\ 1+2 \end{pmatrix} = \begin{pmatrix} 4 \\ 3 \end{pmatrix}, \qquad c - A^\top p = \begin{pmatrix} 3 \\ 2 \\ 3 \\ 2 \end{pmatrix} > 0.$$

$$c^\top x = -3, \qquad b^\top p = -18, \qquad \text{gap} = 15.$$

The products are (3, 2, 6, 4), not all the target value 1. Both points are feasible, so their objective difference bounds the error.

Expand one product before writing a matrix

Write the proposed new values as $x + \Delta x$, $p + \Delta p$, and $r + \Delta r$.

$$(x_j + \Delta x_j)(r_j + \Delta r_j) = x_j r_j + r_j \Delta x_j + x_j \Delta r_j + \Delta x_j \Delta r_j.$$

Newton's method drops the product of the two changes. The resulting equation is linear in the unknown changes:

$$r_j \Delta x_j + x_j \Delta r_j = \mu - x_j r_j.$$

Preserve the two feasible equalities exactly: $A\Delta x = 0$ and $A^\top \Delta p + \Delta r = 0$.

The Newton system collects those same equations

Let $X = \operatorname{diag}(x_1, \dots, x_n)$, $R = \operatorname{diag}(r_1, \dots, r_n)$, and $\mathbf{1}$ be the vector of n ones.

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^\top & I \\ R & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta p \\ \Delta r \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \mu \mathbf{1} - Xr \end{pmatrix}.$$

Block row	Meaning
1	Preserve primal equalities
2	Preserve dual equalities
3	Move the complementarity products toward μ

The zero right-hand sides in the first two rows assume the starting equalities already hold. Positivity must still be checked after solving.

Substitute the numbers into the Newton equations

Order x = (u, v, s_1, s_2). Start: $x = (1, 1, 2, 2)^\top$, $p = (-3, -2)^\top$, $r = (3, 2, 3, 2)^\top$. Target $\mu = 1$.

$$\Delta x_1 + \Delta x_2 + \Delta x_3 = 0, \qquad \Delta x_1 + \Delta x_4 = 0,$$
$$\Delta r_1 = -\Delta p_1 - \Delta p_2, \qquad \Delta r_2 = -\Delta p_1,$$
$$\Delta r_3 = -\Delta p_1, \qquad \Delta r_4 = -\Delta p_2.$$

$$3\Delta x_1 + \Delta r_1 = -2, \qquad 2\Delta x_2 + \Delta r_2 = -1,$$
$$3\Delta x_3 + 2\Delta r_3 = -5, \qquad 2\Delta x_4 + 2\Delta r_4 = -3.$$

For example, the first product equation is $3\Delta x_1 + 1\Delta r_1 = 1 - (1)(3) = -2$.

Eliminate variables to solve two equations

Use the four dual equations in the four product equations. They give:

$$\begin{aligned}\Delta x_1 &= (-2 + \Delta p_1 + \Delta p_2)/3,\\ \Delta x_2 &= (-1 + \Delta p_1)/2,\\ \Delta x_3 &= (-5 + 2\Delta p_1)/3,\\ \Delta x_4 &= (-3 + 2\Delta p_2)/2.\end{aligned}$$

Substitute these expressions into the two primal equations and multiply by 6:

$$9\Delta p_1 + 2\Delta p_2 = 17, \qquad 2\Delta p_1 + 8\Delta p_2 = 13.$$

$$\Delta p_1 = 55/34, \qquad \Delta p_2 = 83/68.$$

Recover every component of the direction

Order x = (u, v, s_1, s_2). Start: $x = (1, 1, 2, 2)^\top$, $p = (-3, -2)^\top$, $r = (3, 2, 3, 2)^\top$. Target $\mu = 1$.

$$\Delta x = \frac{1}{68}\begin{pmatrix} 19 \\ 21 \\ -40 \\ -19 \end{pmatrix}, \quad \Delta p = \frac{1}{68}\begin{pmatrix} 110 \\ 83 \end{pmatrix}, \quad \Delta r = \frac{1}{68}\begin{pmatrix} -193 \\ -110 \\ -110 \\ -83 \end{pmatrix}.$$

For example, $\Delta x_1 = (-2 + 55/34 + 83/68)/3 = 19/68$.

> The first two changes increase u and v. The last two decrease their constraint slacks. Verify: $19 + 21 - 40 = 0$ and $19 - 19 = 0$, so $A\Delta x = 0$.

This is a simultaneous primal–dual direction. It does not choose an entering variable or replace a basis column.

How far can we move while keeping x and r positive?

For a decreasing component, $x_j + \alpha \Delta x_j > 0$ requires $\alpha < -x_j/\Delta x_j$. Apply the same test to r.

$$\widehat{\alpha}_x = \min_{\Delta x_j < 0} \frac{-x_j}{\Delta x_j}, \qquad \widehat{\alpha}_r = \min_{\Delta r_j < 0} \frac{-r_j}{\Delta r_j}.$$

An empty minimum is +∞. A safe step is a fraction of the limiting value, capped at 1; for example, choose a safety factor η = 0.9:

$$\alpha = \min\{1, \eta\widehat{\alpha}_x, \eta\widehat{\alpha}_r\}.$$

> Reaching the boundary is allowed in simplex, but not in a logarithmic-barrier iteration. Backtracking may reduce α further to obtain sufficient progress.

For this direction, a full step is allowed

Order x = (u, v, s_1, s_2). Start: $x = (1, 1, 2, 2)^\top$, $p = (-3, -2)^\top$, $r = (3, 2, 3, 2)^\top$. Target $\mu = 1$.

Positive vector	First boundary step
x	$\min\{2/(40/68), 2/(19/68)\} = 17/5$
r	$\min\{3/(193/68), 2/(110/68), 3/(110/68), 2/(83/68)\} = 204/193$

Both limits exceed 1, so $\alpha = 1$ keeps every component positive. With a safety factor 0.9, the preceding conservative rule would instead give $\alpha = 918/965$.

> For this hand calculation, use the admissible full step α = 1. We checked positivity directly; a full step is not automatically safe in other examples.

The updated point gives a better objective bound

With $x = (u, v, s_1, s_2)^\top$, the full step gives:

$$x^+ = \frac{1}{68}\begin{pmatrix} 87 \\ 89 \\ 96 \\ 117 \end{pmatrix}, \quad p^+ = \frac{1}{68}\begin{pmatrix} -94 \\ -53 \end{pmatrix}, \quad r^+ = \frac{1}{68}\begin{pmatrix} 11 \\ 26 \\ 94 \\ 53 \end{pmatrix}.$$

All $x^+, r^+ > 0$. The primal equalities hold: $87 + 89 + 96 = 4(68)$ and $87 + 117 = 3(68)$.

$$c^\top x^+ = -263/68, \qquad b^\top p^+ = -535/68.$$

$$-535/68 \;\le\; f^* \;\le\; -263/68, \qquad \text{gap} = 4.$$

One Newton step need not land on the central path

For target $\mu = 1$, the linearized equations imply:

$$(x_j + \Delta x_j)(r_j + \Delta r_j) = 1 + \Delta x_j \Delta r_j.$$

We dropped the last product when constructing the step. It does not disappear from the actual updated point.

$$x_1^+ r_1^+ = \frac{87 \cdot 11}{68^2} = \frac{957}{4624} \approx 0.207 \neq 1.$$

> The iterate is feasible and its gap is smaller, but it is not the exact μ = 1 central-path point. More corrections can improve how closely the products match their target.

Here the products happen to sum to 4 because $\Delta x^\top \Delta r = -(A\Delta x)^\top \Delta p = 0$.

Your turn: check a step and an accuracy claim

1. A positive component has $x_j = 0.2$ and direction $\Delta x_j = -0.5$. Find the boundary step and a strictly positive step using η = 0.9.

2. A primal-feasible x has objective –6.98, and a dual-feasible p has objective –7.03. What interval contains the unknown optimum? How far above the optimum can this x be?

Use about 4 minutes. State which feasibility assumptions justify your objective bound.

A short step preserves positivity; feasibility gives the bound

$$\widehat{\alpha} = 0.2/0.5 = 0.4, \qquad \alpha = 0.9(0.4) = 0.36.$$

The updated component is $0.2 - 0.36(0.5) = 0.02 > 0$.

$$-7.03 \le f^* \le -6.98, \qquad 0 \le c^\top x - f^* \le 0.05.$$

The objective difference is a valid error bound because both points are feasible for their respective LPs.

From one correction to a path-following method

The central path is a family of targets. An interior-point algorithm follows those targets approximately:

Operation	Purpose
Start with positive primal/dual variables	Supply an interior iterate; check or handle equality residuals
Choose a smaller positive target μ	Ask for smaller complementarity products
Solve the Newton equations	Compute a correction toward that target
Choose a positive step and check progress	Keep x and r positive and control the approximation
Repeat until feasibility and gap tests pass	Obtain a solution to the requested accuracy

The choice of target, starting neighborhood, and step acceptance rule distinguishes different IPM algorithms. Positivity alone is not a convergence proof.

Appendix: the textbook algorithm with a proved iteration bound.

Why the primal–dual gap bounds the objective error

Assume $Ax = b$, $x \geq 0$, and $r = c - A^\top p \geq 0$. Then:

$$c^\top x - b^\top p = (A^\top p + r)^\top x - p^\top b = r^\top x.$$

$$0 \leq c^\top x - f^* \leq c^\top x - b^\top p = r^\top x.$$

At an exact central-path point, $x_j r_j = \mu$ for all n variables, so the gap is $n\mu$. At a numerical iterate, compute $r^\top x$; do not assume it equals the requested $n\mu$.

In numerical work, check feasibility as well as the gap

A residual is the amount by which an equation or inequality fails. For the standard-form primal–dual pair, monitor:

Check	Quantity that should be small
Primal equalities	$\|Ax - b\|_\infty$
Dual equalities	$\|A^\top p + r - c\|_\infty$
Nonnegativity violations	$\max_j\{0, -x_j, -r_j\}$
Objective difference	$\|c^\top x - b^\top p\|$

The infinity norm $\|v\|_\infty = \max_i |v_i|$ is the largest absolute component. Scale tolerances to the data and objective magnitudes.

> A small objective difference at infeasible points does not certify optimality. The exact error bound on the previous page assumes exact feasibility.

Most work in a Newton step is a linear-system solve

Eliminating Δx and Δr gives a system with one unknown per primal equality:

$$ADA^\top \Delta p = -AR^{-1}(\mu \mathbf{1} - Xr), \qquad D = XR^{-1}.$$

Since x and r are positive, D has positive diagonal entries. If A has independent rows, $ADA^\top$ is positive definite and can be factored to solve the system.

> Sparse means many entries are zero. Exploiting zero entries can reduce work and storage, but matrix products and factorization may create new nonzero entries.

Appendix: derive this system and recover the directions.

What the polynomial-time result actually says

> With an appropriate interior starting point, a specified neighborhood of the central path, and the textbook step rule, the number of iterations needed for gap at most ε is of order $\sqrt{n}\log(n\mu_0/\varepsilon)$.

This is a bound for a particular path-following algorithm under its assumptions. It is not a claim that every damped Newton implementation has that guarantee.

Each iteration still costs work: solve a linear system. Structure can reduce this work, but the savings depend on the matrix.

Crossover means constructing a basic feasible solution from an approximate interior solution, often with simplex steps. It is useful when a basis is needed.

Appendix: theorem assumptions and complete proof.

A different question: can we avoid the Newton solve?

For a very large LP, storing and factoring the Newton system can be expensive. We now introduce a different algorithm with simpler individual iterations.

> **Primal–dual hybrid gradient (PDHG)** alternates updates of primal and dual variables using matrix–vector products and simple bound operations. It does not follow the logarithmic-barrier central path.

Method	Main operation in an iteration
Interior-point method	Solve a Newton linear system
PDHG	Multiply by A and A^T; update vectors

A cheaper iteration may require many more iterations. This is a computational tradeoff, not a universal claim that one method is faster.

One expression connects the primal and dual updates

For $\min\{c^\top x : Ax = b,\ x \geq 0\}$, define the Lagrangian:

$$L(x, p) = c^\top x + p^\top (b - Ax), \qquad x \geq 0, \quad p \text{ unrestricted.}$$

If $Ax = b$, the second term is zero. If $Ax \neq b$, maximizing over unrestricted p can make that term arbitrarily large.

> A **saddle point** minimizes L over x while maximizing it over p. Under LP strong duality with attained optima, finding such a point gives optimal primal and dual solutions.

The two gradients are $c - A^\top p$ for x and $b - Ax$ for p. The updates use descent in x and ascent in p.

PDHG: three explicit updates

Choose positive step sizes τ and σ. Superscript k counts iterations. Start with $x^0 \geq 0$; p is unrestricted.

$$
\begin{aligned}
x^{k+1} &= \left[x^k - \tau(c - A^\top p^k)\right]_+, \\
\bar{x}^{k+1} &= 2x^{k+1} - x^k, \\
p^{k+1} &= p^k + \sigma\left(b - A\bar{x}^{k+1}\right).
\end{aligned}
$$

The operation $[w]_+$ replaces each negative component by zero. The vector $\bar{x}^{k+1}$ looks one more step in the latest primal direction; it is used only in the dual update.

> A standard fixed-step convergence condition is $\tau\sigma\|A\|_2^2 < 1$, assuming a saddle point exists. Here $\|A\|_2$ is the largest factor by which A can increase a vector's Euclidean length.

First PDHG step: update the primal variables

Same LP: minimize $f = -2u - v$, subject to $u + v \leq 4$, $u \leq 3$, $u, v \geq 0$.

Use $x^0 = (1, 1, 2, 2)^\top$, $p^0 = (0, 0)^\top$, and $\tau = \sigma = 1/4$. Unlike the IPM start, this p is not dual feasible.

$$x^1 = \left[\begin{pmatrix} 1 \\ 1 \\ 2 \\ 2 \end{pmatrix} - \frac{1}{4} \begin{pmatrix} -2 \\ -1 \\ 0 \\ 0 \end{pmatrix} \right]_+ = \begin{pmatrix} 3/2 \\ 5/4 \\ 2 \\ 2 \end{pmatrix}.$$

Nonnegativity holds, but $Ax^1 = (19/4, 7/2)^\top \neq (4, 3)^\top$. PDHG does not preserve primal equality feasibility at every iteration.

The step sizes satisfy the sufficient bound because $\|A\|_2^2 = (5 + \sqrt{5})/2$ and $\tau\sigma\|A\|_2^2 \approx 0.226 < 1$.

Then update the dual variables using the equation errors

The previous page gave $x^1 = (3/2, 5/4, 2, 2)^\top$. Recall $Ax = (x_1 + x_2 + x_3,\ x_1 + x_4)^\top$.

$$\bar{x}^1 = 2x^1 - x^0 = \begin{pmatrix} 2 \\ 3/2 \\ 2 \\ 2 \end{pmatrix}, \qquad b - A\bar{x}^1 = \begin{pmatrix} 4 - 11/2 \\ 3 - 4 \end{pmatrix} = \begin{pmatrix} -3/2 \\ -1 \end{pmatrix}.$$

$$p^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \frac{1}{4}\begin{pmatrix} -3/2 \\ -1 \end{pmatrix} = \begin{pmatrix} -3/8 \\ -1/4 \end{pmatrix}.$$

> The negative changes respond to excess left-hand sides in both equality rows. These changed dual variables affect the next primal update through $A^\top p$.

No matrix inverse or Newton system was needed: the operations were products by A and $A^\top$, vector additions, and clipping at zero.

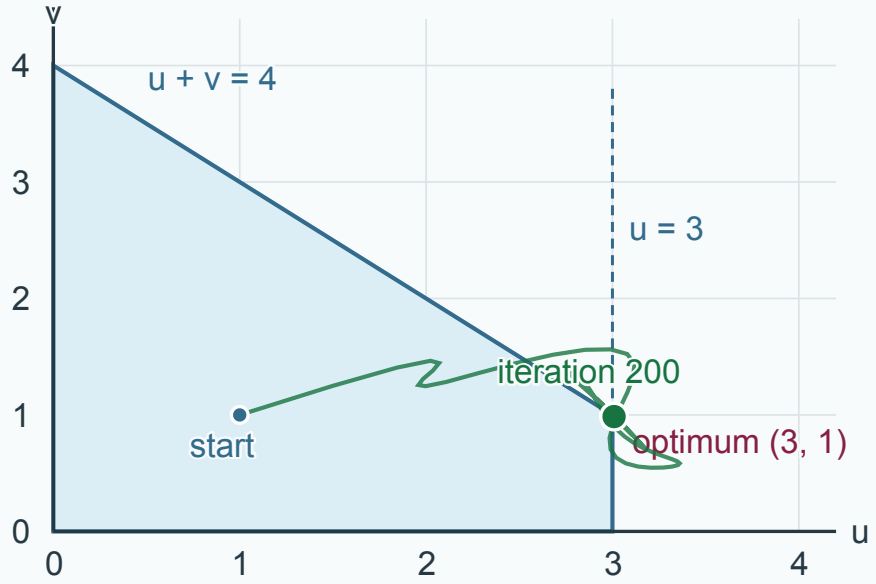
Watch PDHG approach the same LP optimum

Same LP: minimize $f = -2u - v$, subject to $u + v \leq 4$, $u \leq 3$, $u, v \geq 0$.

With $\tau = \sigma = 1/4$, iteration 200 gives $u \approx 3.00873$, $v \approx 0.98587$.

The largest absolute equality error is 0.00873. Objective −7.0033 does not beat the optimum −7 because this point is infeasible.

By iteration 2000, the equality error is below 10^{-10} for this example.



Iteration 200 · x = (3.009, 0.986, 0, 0)
Largest equality error = 0.00873; objective = -7.003.

x order: (u, v, s_1, s_2). The plot shows only (u, v). Equality error checks both equations Ax = b; the objective alone is not an optimality certificate.

The objective can look better because the point is infeasible

For the PDHG iterates, always check $Ax - b$, $x \geq 0$, and the dual inequalities $A^\top p \leq c$.

> An iterate can have objective below –7 because it violates a constraint. That does not improve the LP optimum; the point is not an admissible LP solution.

Equality errors and objective values need not decrease at every step. Use feasibility and optimality tests together, as in the IPM discussion.

The basic PDHG convergence result assumes suitable step sizes and existence of a solution. Infeasibility detection and reliable numerical stopping require additional care.

PDLP adds practical techniques to PDHG

PDLP is a practical LP algorithm built around PDHG. Its additional techniques include rescaling the data, choosing step sizes adaptively, and restarting the iteration.

> Matrix–vector products can use sparse storage and parallel hardware. This makes PDHG-based LP solvers worth considering when matrix factorization is too expensive.

Performance depends on problem structure, scaling, required accuracy, and implementation. Basic PDHG on our small example is not a performance model for a production solver.

Further reading: Practical Large-Scale Linear Programming using Primal-Dual Hybrid Gradient.

Three methods, one LP optimality condition

Method	How it makes progress	What to verify at the end
Simplex	Change a basis; test reduced costs	Primal feasibility and the reduced-cost sign test
Interior point	Follow barrier targets using Newton steps	Primal/dual feasibility and a small gap
PDHG	Alternate vector updates using A and A^T	Primal/dual feasibility and a valid optimality test

> The algorithms differ in their intermediate points and computation. They solve the same primal–dual optimality conditions.

What you should be able to explain

Interior-point methods	PDHG introduction
Explain the barrier and central path	Explain why avoiding factorization can matter
Derive the linearized product equation	Read the three update equations
Check a direction and positive step length	Explain why intermediate feasibility can fail
Use feasible primal/dual values to bound error	Distinguish PDHG from its practical PDLP enhancements

Appendix: derivations, textbook convergence proof, and computational details.

Appendix: derivations and further study

Deriving and solving the Newton system

Starting without feasible equalities

Textbook path following: assumptions, algorithm and proof

Sparse systems, network structure and crossover

Why the PDHG updates have these signs

Derive the smaller primal–dual Newton system

Assume $Ax = b$, $A^\top p + r = c$, and $x, r > 0$. Set $h = \mu \mathbf{1} - Xr$.

$$A\Delta x = 0, \quad \Delta r = -A^\top \Delta p, \quad R\Delta x + X\Delta r = h.$$

$$\Delta x = R^{-1}h + XR^{-1}A^\top \Delta p.$$

Substitute into $A\Delta x = 0$, with $D = XR^{-1}$:

$$ADA^\top \Delta p = -AR^{-1}h = b - \mu AR^{-1}\mathbf{1}.$$

Solve this system for Δp, then recover Δr and Δx. These are the same equations solved by scalar substitution in the worked example.

Why the reduced matrix is positive definite

Assume A has independent rows and $D = \operatorname{diag}(x_j / r_j)$ with $x, r > 0$. For any nonzero vector y,

$$y^\top A D A^\top y = (A^\top y)^\top D (A^\top y) = \sum_j \frac{x_j}{r_j} (A_j^\top y)^2 > 0.$$

Independent rows imply $A^\top y \neq 0$, so at least one positive-weighted square is nonzero.

> Therefore the reduced system has a unique solution. Remove dependent equality rows or use a numerical method that handles them; do not assume every input A has full row rank.

Cholesky factorization writes a positive-definite matrix as $LL^\top$ with triangular L, then solves two triangular systems. It avoids explicitly computing an inverse.

If the starting equalities fail, correct their residuals

A strictly positive x need not satisfy Ax = b. A positive r and unrestricted p need not satisfy $A^\top p$ + r = c.

$$
\begin{aligned}
A\Delta x &= b - Ax, \\
A^\top \Delta p + \Delta r &= c - A^\top p - r, \\
R\Delta x + X\Delta r &= \mu \mathbf{1} - Xr.
\end{aligned}
$$

With a common step α, each equality residual is multiplied by $1 - \alpha$. Positivity and adequate progress still require a suitable step rule.

> An infeasible-start IPM permits nonzero equality residuals while working to reduce them. This does not mean an arbitrary positive start and arbitrary step rule must converge.

Where do positive starting values come from?

The hand example supplied a primal-feasible $x > 0$ and a dual-feasible $r > 0$. A general LP may not supply either point easily, and some LPs have no strictly feasible point in the displayed variables.

> Possible approaches include auxiliary feasibility problems, removing variables forced to zero, and formulations that solve for feasibility and optimality together. A homogeneous embedding is one such augmented formulation.

These methods also need to distinguish an infeasible problem from an unbounded problem. Small residuals at one arbitrary iterate do not establish either conclusion.

The textbook convergence theorem below assumes an interior starting point close enough to the central path. Its initialization is an assumption, not an automatic consequence of writing an LP in standard form.

The textbook algorithm uses a controlled barrier decrease

Assume A has independent rows; $Ax^0 = b$, $A^\top p^0 + r^0 = c$, and $x^0, r^0 > 0$. Choose $\mu_0 > 0$ and $0 < \beta < 1$ such that

$$\left\| \mu_0^{-1} X^0 r^0 - \mathbf{1} \right\|_2 \le \beta.$$

The Euclidean norm condition bounds the total mismatch between the normalized products and 1.

$$q = 1 - \frac{\sqrt{\beta} - \beta}{\sqrt{\beta} + \sqrt{n}}, \qquad \widehat{\mu} = q\mu.$$

The symbol q is the factor for decreasing μ, not the step length for a line search. It is chosen close enough to 1 to permit the full Newton update on the next page.

The textbook primal path-following update

At the current x, solve for the primal direction d and a new dual vector p^+:

$$\widehat{\mu} X^{-2} d - A^\top p^+ = \widehat{\mu} X^{-1} \mathbf{1} - c, \qquad Ad = 0.$$

$$x^+ = x + d, \qquad r^+ = c - A^\top p^+, \qquad \mu^+ = \widehat{\mu}.$$

> This is the equality-constrained Newton equation for the primal barrier objective. It is a particular algorithm used in the textbook proof; it is not identical to every practical primal–dual predictor–corrector method.

Repeat the controlled decrease and update until the feasible primal–dual gap is at most ε. The following proof shows that these full steps preserve positivity and the neighborhood condition.

Proof: decreasing μ enlarges the mismatch in a controlled way

Suppose $\|\mu^{-1}Xr - \mathbf{1}\|_2 \le \beta$. With $\widehat{\mu} = q\mu$, the triangle inequality gives

$$\left\|\widehat{\mu}^{-1}Xr - \mathbf{1}\right\|_2 \le \frac{\beta}{q} + \frac{1-q}{q}\sqrt{n} = \sqrt{\beta}.$$

The final equality follows from $q = (\sqrt{n} + \beta)/(\sqrt{n} + \sqrt{\beta})$.

> The smaller target temporarily increases the mismatch from at most β to at most √β. The Newton step must restore the stronger bound β.

Proof: bound the relative size of the Newton direction

Multiply the Newton equation by $d^\top$. Use $Ad = 0$ and $c = A^\top p + r$:

$$\|X^{-1}d\|_2^2 = -\left(\widehat{\mu}^{-1}Xr - \mathbf{1}\right)^\top X^{-1}d.$$

The Cauchy–Schwarz inequality and the previous mismatch bound imply

$$\|X^{-1}d\|_2^2 \le \sqrt{\beta}\,\|X^{-1}d\|_2 \quad \Rightarrow \quad \|X^{-1}d\|_2 \le \sqrt{\beta} < 1.$$

If the norm is zero, the conclusion already holds; otherwise divide by the positive norm. Every relative component d_j/x_j consequently has absolute value below 1.

Proof: the new primal and dual points remain positive

Write $t_j = d_j/x_j$. The preceding page gives $|t_j| < 1$. Then

$$x_j^+ = x_j(1 + t_j) > 0.$$

Rearrange the Newton equation to obtain the new dual slack:

$$r^+ = \widehat{\mu} X^{-1}(\mathbf{1} - X^{-1}d), \qquad r_j^+ = \frac{\widehat{\mu}}{x_j}(1 - t_j) > 0.$$

Also $Ax^+ = Ax + Ad = b$, and $A^\top p^+ + r^+ = c$ by definition. Both primal and dual feasibility are preserved.

Proof: the Newton step restores the neighborhood

$$\frac{x_j^+ r_j^+}{\widehat{\mu}} - 1 = (1 + t_j)(1 - t_j) - 1 = -t_j^2.$$

Therefore, using $\sqrt{\sum_j t_j^4} \le \sum_j t_j^2$,

$$\left\| \widehat{\mu}^{-1} X^+ r^+ - \mathbf{1} \right\|_2 = \sqrt{\sum_j t_j^4} \le \|X^{-1} d\|_2^2 \le \beta.$$

This completes the induction: the next iterate is strictly feasible and satisfies the same neighborhood condition.

Proof: count the barrier reductions needed for a small gap

The neighborhood bound implies $n\mu_k(1-\beta) \le (x^k)^\top r^k \le n\mu_k(1+\beta)$. Since $\mu_k = q^k\mu_0$, it suffices that

$$q^k n\mu_0(1+\beta) \le \varepsilon.$$

Use $q^k \le e^{-k(1-q)}$. A sufficient number of iterations is

$$k \ge \frac{\sqrt{\beta}+\sqrt{n}}{\sqrt{\beta}-\beta}\log\frac{n\mu_0(1+\beta)}{\varepsilon}.$$

Take the ceiling, or zero if the initial gap already passes. For fixed β this is $O(\sqrt{n}\log(n\mu_0/\varepsilon))$. Counting arithmetic work also requires the cost of each linear-system solve.

Sparse factorization: where the computation can grow

For the primal–dual step, form $M = ADA^\top$, where $D_{jj} = x_j / r_j$. Its entry is

$$M_{ik} = \sum_j A_{ij} D_{jj} A_{kj}.$$

Rows i and k can interact whenever they use a common column of A. Thus sparse A does not guarantee sparse M. Factoring M can create additional nonzeros, called fill-in.

> Reordering the rows and columns before factorization can reduce fill-in. The benefit depends on the actual pattern; there is no universal sparse per-iteration cost determined only by m and n.

For dense A, forming M costs $O(m^2n)$ arithmetic operations and a dense factorization costs $O(m^3)$.

The same network structure also helps interior-point solvers

For a node–arc incidence matrix, each arc column has +1 at its tail and −1 at its head. Therefore the off-diagonal entry of ADA^T links two nodes only when an arc joins them.

> This is a weighted graph Laplacian: each diagonal is the sum of incident positive weights, and each off-diagonal is minus the total weight joining the two nodes.

Delete one redundant balance row in a connected network to obtain a positive-definite reduced matrix. Graph structure can then guide sparse solution methods.

Assignment problems have related incidence structure. Block-structured LPs can also permit smaller intermediate solves, but the actual speedup depends on the connections between blocks.

An interior solution is not automatically a basis

Near optimality, some primal variables and some dual slacks are small. This can suggest which constraints are active, but numerical size alone does not identify an invertible optimal basis.

> Crossover constructs and improves a basis, typically using simplex operations. Check its independent columns, primal feasibility, and reduced-cost optimality conditions.

Degeneracy can make several bases correspond to one optimum. If a whole edge is optimal, a central-path limit can lie inside that edge rather than at a vertex.

Network integrality guarantees an integral optimal basic solution for integer data; it does not imply every approximate interior iterate is integral.

Practical IPM variants balance progress and centering

A predictor direction can aim toward zero complementarity. A corrector direction compensates for the product terms omitted by linearization and aims for more balanced positive products.

$$(x_j + \Delta x_j)(r_j + \Delta r_j) = x_j r_j + r_j \Delta x_j + x_j \Delta r_j + \Delta x_j \Delta r_j.$$

The last product explains why a second correction can be useful. Implementations also choose step lengths and target parameters based on the computed directions.

> The textbook proof establishes a guarantee for its specified method. Practical predictor–corrector choices need their own analysis; do not transfer the theorem to an arbitrary update rule.

Derive the PDHG signs from the Lagrangian

$$L(x,p) = c^\top x + p^\top (b - Ax).$$

With p fixed, the gradient in x is $c - A^\top p$. A descent step subtracts this gradient, then projects onto $x \geq 0$:

$$x^+ = [x - \tau(c - A^\top p)]_+.$$

With $\bar{x} = 2x^+ - x$, the gradient in p is $b - A\bar{x}$. The maximization step adds this gradient:

$$p^+ = p + \sigma(b - A\bar{x}).$$

The extrapolated x is part of the analyzed PDHG rule. Alternating ordinary descent/ascent without this correction is a different algorithm.

Why a saddle point solves the primal and dual LPs

A saddle point satisfies $L(x^*, p) \le L(x^*, p^*) \le L(x, p^*)$ for every unrestricted p and every x ≥ 0.

The left inequality forces $Ax^* = b$: otherwise choose $p = p^* + t(b - Ax^*)$ and let t grow. The right inequality forces $r^* = c - A^\top p^* \ge 0$: a negative component would make L decrease without bound as its x component increases.

Minimizing $(r^*)^\top x + b^\top p^*$ over x ≥ 0 also gives $(r^*)^\top x^* = 0$. Thus the primal and dual are feasible with equal objective values.

> Conversely, any optimal primal–dual pair with zero complementarity satisfies both saddle inequalities. This connects PDHG to the same LP optimality conditions used by simplex and IPM.