The ASM Lab ventilation solver

How the airflow solver, the linear algebra and the coupled airflow–thermal solver work, with worked examples computed step by step

ASM Lab 2026.1.12 · Technical reference · October 2026

Contents

Part A — The airflow solver **5**
A1. What the solver computes 5
A2. The network as a graph 5
A3. Airway resistance 6
A4. Air density 7
A5. Elevation and natural ventilation 8
A6. Fans 9
A7. The equations: Kirchhoff's laws for air 10
A8. Newton–Raphson iteration 11
A9. The Jacobian 11
A10. The zero-flow singularity and the slope cap 12
A11. Line search, tolerance and stopping 12
A12. Summary of one iteration 13
A13. Example 1 — two parallel drives (case02) 13
A14. Example 2 — a fan on a curve (case05) 14
A15. Example 3 — the balanced Wheatstone bridge (lit2) 18
A16. Example 4 — natural ventilation (case09) 24
A17. Larger networks 28

Part B — Solving the linear systems **30**
B1. The airflow system 30
B2. Gaussian elimination and the LU factorisation 30
B3. The Cholesky factorisation 30
B4. Sparsity, fill-in and the envelope 32
B5. Why a direct solve 33
B6. Conjugate gradients with incomplete Cholesky (fall-back) 33
B7. The coupled system is not symmetric 34
B8. Why GMRES 35
B9. How GMRES works 35

Part C — The coupled airflow and thermal solver **38**
C1. Why airflow and heat must be solved together 38
C2. The state of the air 39
C3. Heat along an airway 40
C4. Mixing at nodes 43
C5. The coupled iteration 44
C6. Worked example — natural ventilation with real temperatures (case09) 46
C7. Convergence on the sample mines 50
C8. The all-at-once Newton formulation (in the engine, not the default) 51

Part D — The fire simulation **54**

D1. What it computes 54
D2. Plug flow along an airway 54
D3. Mixing at junctions 55
D4. The fire 55
D5. Heat lost to the walls 56
D6. The airflow answers the fire 57
D7. Reading a run 58
D8. Smoke upstream: the critical velocity 58
D9. Worked example — a 5 MW fire in a drift 59
D10. How far can it be trusted? 60

Part E — The CFD solver **62**
E1. What it is for 62
E2. The mesh: the section swept along the path 62
E3. The equations 63
E4. The finite-volume method 63
E5. Turbulence and the wall 64
E6. Roughness: from Atkinson's k 65
E7. What comes out 66
E8. Worked example — a straight tunnel 66

Introduction

This document explains how ASM Lab computes airflow and air conditions in a mine ventilation network. It is written to be read alongside the program: every equation is the one the program uses, and every worked example is taken from the program's own test networks, with the numbers recorded from the solver as it ran. Where a calculation is shown by hand, the same figures can be reproduced on a calculator or in a spreadsheet.

Part A covers the airflow solver: the airway laws (Atkinson resistance, density, natural ventilation, fans, regulators), the nodal formulation of Kirchhoff's laws, Newton–Raphson iteration, the Jacobian, line search and convergence, and what the program adds around that core: the air state from pressure, dry bulb and wet bulb, the gravity and density-basis settings, fans corrected for air density, reverse flow and fixed-quantity fans, and walls. It then follows the solver iteration by iteration through four networks: parallel drives, a fan on its curve, a Wheatstone bridge and a natural-ventilation loop.

Part B covers the linear systems solved at every iteration: Gaussian elimination and LU, the Cholesky factorisation, sparse envelope storage and the reverse Cuthill–McKee ordering, conjugate gradients with incomplete Cholesky, and, for the non-symmetric all-at-once coupled system (kept in the engine, not the default), the LU + GMRES combination and why GMRES is used.

Part C covers the coupled airflow and thermal solver, which settles the airflow and the heat in turn and relaxes the air between rounds (Aitken, then exchange and Anderson acceleration where the rounds stall): the thermal model (autocompression, Gibson rock heat with the plain and the film wall models, heat and water from sources, coolers, mixing, still air), the coupled iteration, a full worked example with hand checks of the heat transfer, the natural ventilation pressure and the energy balance, its convergence on the sample mines, and, last, the all-at-once Newton formulation that the program no longer runs.

Part D covers the fire simulation: plug flow along the airways, mixing at junctions, the three fire types and their heat, fume and gases, the classic and the tested models of heat lost to the walls, the way the airflow answers the fire's heat, how a run is read, and a worked example with hand checks, followed by what measurement says about how far it can be trusted. **Part E** covers the CFD solver: the body-fitted mesh swept along the tunnel, the steady incompressible equations on it by SIMPLE with Rhie–Chow fluxes, the k–epsilon and mixing-length turbulence models, the wall functions and the roughness link to Atkinson's factor, with a worked straight-tunnel example against the Colebrook–White law.

Units are SI throughout: pressure in Pa, flow in m^3/s, mass flow in kg/s, enthalpy in kJ per kg of dry air, moisture in g or kg of water per kg of dry air, resistance in $N \cdot s^2/m^8$. Gravity is a project setting, 9.79 m/s^2 by default (McPherson's field book).

Part A — The airflow solver

A1. What the solver computes

A mine ventilation network is a set of *airways* (drives, shafts, raises, crosscuts, ducts) joined at *nodes* (junctions, shaft stations, portals). Air moves through it because something creates a pressure difference: a fan, the weight difference between a cold downcast and a warm upcast column of air (natural ventilation), or a pressure imposed at the surface connections. The solver's job is to find, for a given network, the steady state in which every airway carries the flow its pressure difference drives and no node gains or loses air.

The result is the **pressure at every node** and the **volume flow, mass flow and pressure drop in every airway**, plus the fan operating points, regulator settings and the air density everywhere. Everything else the program shows (velocities, power, colour maps, compliance checks) is derived from these.

ASM Lab solves the network by the **nodal pressure method** with **Newton–Raphson** iteration: the unknowns are the pressures at the free nodes, and the equations are conservation of mass at each of those nodes. This part explains each ingredient, from the airway law to convergence, and then follows the solver step by step through four small networks, with every number written out so it can be checked by hand.

Notation

Table 1. Notation used in Part A.

Symbol	Meaning	Unit
P_i	absolute static pressure at node i	Pa
q_e	volume flow in airway e, positive from its source node to its target node	m³/s
ρ_e	air density in airway e	kg/m³
R_e	Atkinson resistance of airway e at its operating density	N·s²/m⁸
z_i	elevation of node i	m
Δ_e	effective pressure difference driving airway e	Pa
F_i	mass residual (imbalance) at free node i	kg/s
J	Jacobian matrix, J_ij = ∂F_i/∂P_j	kg/(s·Pa)
g	gravitational acceleration: a project setting (Settings → Engineering), default 9.79 (McPherson); only 9.7–9.9 is accepted	m/s²

A2. The network as a graph

The network is held as a directed graph. Each airway has a *source* and a *target* node; the direction only fixes the sign convention (a negative flow simply runs the other way). Nodes are of two kinds:

- **Boundary (fixed-pressure) nodes**: surface intakes and exhausts, portals, or any node given a pressure. Their pressure is known and never changes.

- **Free nodes**: every other junction. Their pressure is unknown; there is one equation (mass balance) for each.

Figure 1 shows the Wheatstone bridge test case used later (lit2): four nodes, five airways, one intake A and one exhaust D. B and C are free nodes; the diagonal BC connects the two sides.

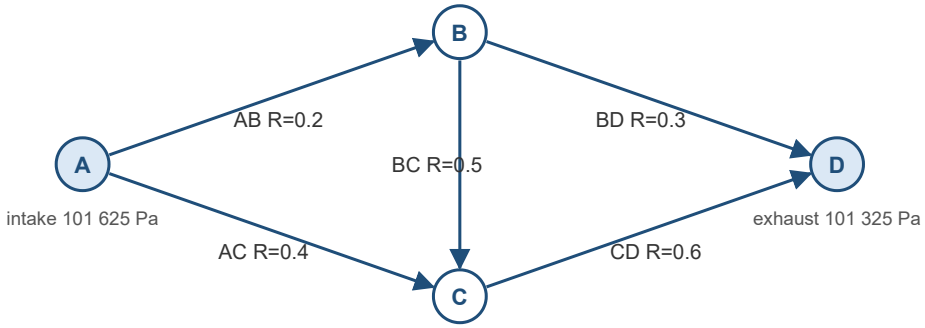


Figure 1. The balanced Wheatstone bridge (lit2). Shaded: fixed-pressure nodes. Arrows show each airway's source → target direction.

Dead ends. A free node with only one airway can balance its mass only at zero flow, so that airway carries nothing. Before solving, ASM Lab peels such stubs off repeatedly (removing one may expose the next along the same spur) and reports them at zero flow. An airway with a fan or regulator on it is never removed silently: a device on a dead end is a modelling error the user must see. Pruning keeps the linear system smaller and stops the tiny, noisy imbalance of a stub from deciding convergence.

A3. Airway resistance

The pressure lost to friction in an airway follows the **square law** of Atkinson:

$$\Delta P = R\,Q\,|Q| \tag{A1}$$

The $Q|Q|$ form keeps the sign: air flowing backwards loses pressure in the backward direction. From the geometry, at the reference density $\rho_{\mathrm{ref}} = 1.2$ kg/m³,

$$R_{\mathrm{std}} = \frac{k\,L\,\mathrm{Per}}{A^3} + \frac{X\,\rho_{\mathrm{ref}}}{2A^2} \tag{A2}$$

where k is the Atkinson friction factor (kg/m³), L the length, Per the perimeter, A the cross-sectional area and X the dimensionless shock-loss factor (bends, junctions, obstructions: $\Delta P_{\mathrm{shock}} = X\rho V^2/2$). Rectangular, arched (semicircular roof) and circular profiles are supported, and a stated area and perimeter (an irregular profile) override the shape formulas. A user can also give R directly (a resistance override). A **wall** (a stopping, door or brattice) is its resistance and nothing else: $\Delta P = R\,Q|Q|$ with the wall's own R, whatever the

airway's length, size or friction factor; a wall that is blocked, like an airway marked as not ventilated, is left out of the solve and carries no air.

Worked example (case02). A 5 m × 5 m drive, 100 m long, $k = 0.01$ kg/m³, no shock loss: $A = 25$ m², Per = 20 m, so

$$R_{\text{std}} = \frac{0.01 \times 100 \times 20}{25^3} = \frac{20}{15\,625} = 1.280 \times 10^{-3}\ \text{N s}^2/\text{m}^8$$

Minimum resistance. A zero resistance is not a low-loss airway but a degenerate one: its two ends are the same point and the pressure difference across it is undefined. Every R is floored at $R_{\min} = 10^{-6}$ N·s²/m⁸, which at 100 m³/s costs only 0.01 Pa. (Other ventilation programs write 0 on fan branches as a convention; the floor reads it as “negligible”.)

Density scaling. R as given is for standard air. Friction and shock losses scale with density, so the solver uses $R = R_{\text{std}}\,\rho_e/\rho_{\text{ref}}$ (unless the resistance is locked). In case02 the airway density is 1.19950 kg/m³ (Section A4), giving $R = 1.280 \times 10^{-3} \times 1.19950/1.2 = 1.2795 \times 10^{-3}$, exactly the value in the solver's trace (Section A13).

Regulators are modelled as an extra resistance from their opening area A_r (Blackbook eq. 2.18; McPherson $A_r = 1.2\,Q\sqrt{\rho/\Delta P}$):

$$R_{\text{reg}} = \frac{1.44\,\rho}{A_r^2} \tag{A3}$$

A regulator set to a target pressure drop has its area found from the flow it carries; a fixed-quantity airway is treated as a current source (its flow is prescribed and does not depend on pressures), and reports the pressure the network needs it to supply. Where that pressure is a rise rather than a loss, the airway is acting as a fan: it is reported as one, and its shaft power is heat in Part C.

A4. Air density

Mass, not volume, is conserved at a junction, so the solver needs the density of the air in every airway. It is computed from the pressure, the dry-bulb temperature and the moisture at each node, using the psychrometric relations of the program:

$$e_s(t) = 610.5\ \exp\!\left(\frac{17.27\,t}{237.3 + t}\right)\ \text{Pa} \quad \text{(Tetens; Blackbook eq. 5.6)} \tag{A4}$$

$$p_w = e_s(t_{wb}) - \gamma\,P\,(t_{db} - t_{wb}), \qquad \gamma = 6.44 \times 10^{-4}\ \text{K}^{-1} \tag{A5}$$

$$\rho = \frac{P - 0.378\,p_w}{287.04\,(t_{db} + 273.15)} \tag{A6}$$

where p_w is the partial pressure of water vapour. Moist air is lighter than dry air at the same pressure and temperature, which the $0.378\,p_w$ term accounts for.

How the air at a node is stated. In the program the air is always given by the barometric pressure, the dry bulb and the wet bulb, and p_w follows from Eq. A5; the relative humidity is

only ever worked out from them and reported, never assumed. (The core also accepts a relative humidity in place of the wet bulb, $p_w = \mathrm{RH} \cdot e_s(t_{db})$; and a core node with no stated air at all is taken at 20 °C and 50 % RH, which the four worked examples below use.) The application never leaves a node without air. A node that states nothing takes the *site's surface air* (Settings → Thermal: dry bulb, wet bulb and pressure at the surface), carried down or up to the node's elevation z at the same dry bulb and the same moisture content:

$$P(z) = P_s \exp\left(\frac{-g\,(z - z_s)}{287.05\,(273.15 + t_{db})}\right) \tag{A6a}$$

where $r_s = 0.622\,p_w/(P_s - p_w)$ is the surface air's moisture content (kg of vapour per kg of dry air) and z_s the surface elevation (the stated collar, else the highest intake). A node that states a dry bulb but no wet bulb keeps its dry bulb and takes the surface air's moisture. The wet bulb at the node is then that of air with moisture r_s at $P(z)$ and the dry bulb t_{db}. No humidity is assumed anywhere in the chain.

Density basis. Eq. A6 is the density of the moist air as a whole, the default (“moist”). A project can instead choose the “dry” basis (Settings → Engineering), the density of the dry air in the mixture, which is what some other programs report and scale their fans with:

$$\rho_{\mathrm{dry}} = \frac{P - p_w}{287.04\,(t_{db} + 273.15)} \tag{A6b}$$

The choice moves every density-dependent quantity together (airway resistance, column weight, fan pressure, the mass flow that is conserved), by the factor $\rho_{\mathrm{dry}}/\rho \approx 1 - 0.622\,p_w/P$, about 1 % at typical mine conditions, and it is applied before each solve like the gravity setting.

Worked example (case02). Node A is at 101 425 Pa and node B at 101 325 Pa, both 20 °C and 50 % RH. $e_s(20) = 610.5\,e^{17.27\times 20/257.3} = 2337.1$ Pa, so $p_w = 0.5 \times 2337.1 = 1168.6$ Pa, and

$$\rho_A = \frac{101\,425 - 0.378 \times 1168.6}{287.04 \times 293.15} = 1.20010, \qquad \rho_B = \frac{101\,325 - 0.378 \times 1168.6}{287.04 \times 293.15} = 1.19891\ \mathrm{kg/m^3}$$

The airway density is the mean of its two end nodes: $\rho_e = (1.20010 + 1.19891)/2 = 1.19951$ kg/m³ — the solver's value is 1.19950 (the small difference comes from solving the wet bulb back from the humidity and recomputing p_w by Eq. A5).

Because density depends on pressure, and pressure is the unknown, the densities are **refreshed at the start of every Newton iteration** and again after each accepted step (so convergence is judged on the state that will be reported). After the solve the program reports the density spread; above 5 % the air can no longer be treated as having one density and a warning recommends running with real node temperatures (the coupled solver of Part C).

A5. Elevation and natural ventilation

A column of air has weight. Between two nodes at different elevations, the pressure difference that drives flow is not $P_s - P_t$ but that difference corrected for the weight of the air

column between them:

$$\Delta_e = P_s - P_t - \rho_e\, g\,(z_t - z_s) \tag{A7}$$

For a downcast shaft ($z_t < z_s$) the correction is positive: air flows down a shaft even with equal pressures at both ends if nothing resists it, and the pressure at the bottom is higher by $\rho g H$. Natural ventilation is the consequence: in a closed loop with a cold downcast and a warm upcast, $\sum \rho_e g\, \Delta z_e$ around the loop does not cancel, and the difference, the **natural ventilation pressure**,

$$\mathrm{NVP} = g\, H\, (\bar{\rho}_{\mathrm{down}} - \bar{\rho}_{\mathrm{up}}) \tag{A8}$$

drives air round the loop like a fan. In the solver no special term is needed: Eq. A7 applied to every airway, with each airway's own density, produces it automatically. Section A16 (case09) shows it. The same g is used everywhere it appears: the column weight above, the starting pressures of the iteration (Section A8), the barometric relation of Eq. A6a and the autocompression of Part C. It is the project's gravity setting (default 9.79 m/s²) and is fixed before each solve.

A6. Fans

A fan is an airway that adds pressure. Its law combines the airway's own resistance with the fan characteristic $P_{\mathrm{fan}}(q)$:

$$R\, q\, |q| - P_{\mathrm{fan}}(q) = \Delta_e \tag{A9}$$

Two forms of characteristic are used:

- **Quadratic:** $P_{\mathrm{fan}} = a - b\, q|q|$. Then Eq. A9 becomes $(R + b)\, q|q| = \Delta_e + a$ and is solved in closed form like an ordinary airway with $R_{\mathrm{eff}} = R + b$.
- **Tabulated curve** (manufacturer points, linearly interpolated, scaled by ρ/ρ_{ref}): on each segment $P_{\mathrm{fan}} = Mq + B$, so Eq. A9 is the quadratic $Rq^2 - Mq - (B + \Delta_e) = 0$. Every segment's roots are computed exactly and the root closest to the previous iterate is kept, which avoids the fan jumping between operating points (and keeps it out of the stall zone where possible).

The fan's slope dP_{fan}/dq enters the Jacobian (Section A9), so a steep fan curve makes the fan's flow insensitive to the network pressures, exactly as in reality.

Air density and the curve. A manufacturer's curve is for one air density, ρ_{ref} (1.2 kg/m³ unless the curve says otherwise). By the fan laws the pressure at another density is in proportion:

$$P_{\mathrm{fan}}(q;\rho) = \frac{\rho}{\rho_{\mathrm{ref}}}\, P_{\mathrm{ref}}(q) \tag{A9a}$$

ρ is the density of the air in the fan's own airway, refreshed with the rest of the thermodynamic state at every iteration. The curve can also be used *as entered* at any density (untick “Correct the curve for air density” on the fan, or set it for the project in

Settings → Engineering): then $\rho/\rho_{\text{ref}} = 1$. A project may hold the correction at the surface design air instead of the air at each fan, which keeps a fan's duty the same whatever its depth. The tabulated roots of the previous paragraph are always found on the curve at the density actually used, the same one the residual and the Jacobian use; solving them on the raw points while the residual used the scaled curve left the two disagreeing by about 17 % at 1.4 kg/m³ and is the reason Example 2 below now gives 53.57 m³/s and not 53.45.

Reverse flow. A running fan forced backwards by the rest of the network does not follow the mirror image of its curve: the impeller keeps pushing the same way. For $q < 0$ the fan therefore still adds its *shut-off* pressure $P_0 = P_{\text{fan}}(0)$, forward:

$$P_{\text{fan}}(q) = P_0 \quad (q < 0), \qquad \text{so} \qquad q = -\sqrt{\frac{-(P_0 + \Delta_e)}{R}} \ \text{ when } P_0 + \Delta_e < 0 \tag{A9b}$$

A fan reverses only when the pressure across it overpowers its shut-off pressure. The mirror-image curve used before gave a second, false operating point that turned a whole circuit round: on a benchmark network a fan that a reference program puts at 57.8 kg/s forward came out at −58.7 kg/s.

Fixed-quantity and fixed-pressure fans. A fan with a fixed quantity is a flat curve that drops almost vertically to 0 Pa at its quantity. It can only add pressure, so where the network pushes more than that quantity through it the fan would have to take pressure out; the point lies past the curve's end and the solver settles on the reverse root, running backwards at minus the quantity. The program therefore continues the drop by the same distance below 0 Pa (a throttle), so the quantity is held from both sides. A fixed-pressure fan is the horizontal curve $P_{\text{fan}} = \text{const}$.

A7. The equations: Kirchhoff's laws for air

Two laws govern a network, the ventilation equivalents of Kirchhoff's laws for electrical circuits:

1. **Mass conservation at a node** (Kirchhoff's current law): the mass of air entering a node equals the mass leaving it, $\sum_{\text{in}} \rho_e q_e = \sum_{\text{out}} \rho_e q_e$.
2. **Pressure around a loop** (Kirchhoff's voltage law): around any closed loop the pressure drops, fan pressures and natural ventilation sum to zero.

Classical hand methods (Hardy Cross) take the airway flows as unknowns, satisfy law 1 by construction and iterate on law 2 loop by loop. ASM Lab instead takes the **node pressures** as unknowns. Then law 2 holds automatically: any loop's pressure drops are differences of node pressures, which telescope to zero. Only law 1 remains, one equation per free node:

$$F_i(P) = \sum_{e:\, t(e)=i} \rho_e\, q_e(P) - \sum_{e:\, s(e)=i} \rho_e\, q_e(P) = 0, \qquad i \in \text{free nodes} \tag{A10}$$

with each airway's flow given by its own law, inverted:

$$q_e = \operatorname{sign}(\Delta_e)\sqrt{\frac{|\Delta_e|}{R_e}} \tag{A11}$$

The nodal form has practical advantages over loop methods: no loop set has to be chosen (a mine with thousands of airways has thousands of independent loops), fixed-pressure boundaries at many surface connections are natural, and the linear systems are symmetric and sparse, which Part B exploits. Its one difficulty is visible in Eq. A11: near zero flow the square root is infinitely steep. Section A10 explains how the solver deals with that.

A8. Newton–Raphson iteration

Eq. A10 is a system of nonlinear equations $F(P) = 0$. Newton's method replaces it, at the current estimate $P^{(k)}$, by its linearisation

$$F(P^{(k)} + \delta P) \approx F(P^{(k)}) + J(P^{(k)})\,\delta P \tag{A12}$$

and chooses the step that makes the linearisation zero:

$$J(P^{(k)})\,\delta P = -F(P^{(k)}), \qquad P^{(k+1)} = P^{(k)} + \alpha\,\delta P \tag{A13}$$

where $\alpha \le 1$ is chosen by a line search (Section A11). Near the solution Newton's method converges **quadratically**: the number of correct digits roughly doubles each iteration. This is visible in every example below: the residual falls slowly while the estimate is far away, then drops by orders of magnitude in the last two or three iterations.

One-dimensional illustration

For a single free node (case05, Section A14) the method is the familiar tangent construction: $\delta P = -F/F'$. Each step follows the tangent of the mass-imbalance curve $F(P)$ to where it crosses zero.

A9. The Jacobian

Differentiating Eq. A10, only airways touching node i contribute to row i. For airway e from s to t, $\partial\Delta_e/\partial P_s = 1$ and $\partial\Delta_e/\partial P_t = -1$, and from Eq. A9 by implicit differentiation,

$$\frac{\partial q_e}{\partial \Delta_e} = s_e = \frac{1}{|\,2R_e|q_e| - dP_{\text{fan}}/dq\,|} \tag{A14}$$

(for a plain airway simply $1/(2R|q|)$). With densities held at their values for the iteration, the entries are

$$J_{ii} = -\sum_{e \ni i} \rho_e s_e, \qquad J_{ij} = \sum_{e:\, i \leftrightarrow j} \rho_e s_e \quad (i \neq j) \tag{A15}$$

Airways to a fixed-pressure node contribute to the diagonal only, since the boundary pressure is not an unknown. Three properties follow, and Part B relies on all of them:

- **Symmetric:** $J_{ij} = J_{ji}$, since the same airway gives both entries.

- **Negative definite (diagonally dominant):** each diagonal is minus the sum of the positive slopes at the node, so $|J_{ii}| \geq \sum_{j \neq i} J_{ij}$, with strict inequality at any node connected to a boundary. Hence $B = -J$ is a symmetric positive definite M-matrix.
- **Sparse:** row i has nonzeros only for the neighbours of i, typically 3 or 4 in a mine.

A **fixed-flow airway** (a fixed-quantity regulator, or a booster modelled as a flow) has $s_e = 0$: its flow does not depend on pressures, so it adds its prescribed ρq to the residual but nothing to the Jacobian, exactly as a current source in circuit analysis.

A10. The zero-flow singularity and the slope cap

Eq. A14 has a problem at $q = 0$: $s = 1/(2R|q|)$ is infinite. Physically, the square-root law $q = \sqrt{\Delta/R}$ has a vertical tangent at $\Delta = 0$. The solver caps the denominator at 10^{-6}, so $s \leq 10^{6}$ m³/s per Pa. This happens in two situations:

- **At the start:** if the initial pressures make two connected nodes equal, the airway between them has zero flow and a slope of 10^{6}. The Newton step is then tiny (Section A15, iteration 1). After one or two small steps the flows become nonzero, the slopes realistic, and Newton proceeds normally.
- **At the solution:** an airway whose true flow is (nearly) zero, like the diagonal of a balanced bridge. Here a full Newton step from $+\Delta$ lands on $-\Delta$: for $q = \sqrt{\Delta/R}$ the secant across zero is exactly twice the tangent, so the iterate jumps to the mirror image and back. The Armijo line search (next section) prevents this period-2 cycle by requiring a real decrease, which forces a half step that lands inside the true solution's basin.

A11. Line search, tolerance and stopping

Armijo line search. A full Newton step can overshoot when the estimate is far from the solution. The solver first tries $\alpha = 1$; it accepts the step if the largest nodal residual falls by a sufficient amount,

$$\max_i |F_i(P + \alpha\,\delta P)| \leq (1 - c\,\alpha)\ \max_i |F_i(P)|, \qquad c = 0.01 \tag{A16}$$

and otherwise halves α, up to 10 times ($\alpha \approx 0.002$). If no trial passes, the last (smallest) step is taken anyway, which guarantees progress through the zero-flow situation of the previous section. Requiring a *sufficient* decrease rather than any decrease is what breaks the period-2 cycle: on a 200-rung ladder test the old “accept if not worse” rule ran 2000 iterations without converging; the Armijo rule converges in 14.

Tolerance. The solve has converged when the largest nodal mass imbalance is below

$$\text{tol} = \max(10^{-4}\ \text{kg/s},\ 10^{-6} \times \max_e |\rho_e q_e|) \tag{A17}$$

i.e. one part in a million of the largest mass flow in the network, but never tighter than 0.1 g/s. A fixed absolute tolerance fails both ways: it asks a 600 kg/s mine for one part in six

million (close to the round-off floor of the summation itself) while letting a small network pass with a badly wrong flow. The tolerance is re-evaluated every iteration, because the first estimate can carry flows far larger than the solution.

Stall and precision limit. The solver also stops if the residual has not improved by 0.1 % in 40 iterations (it then reports the node that dominated the residual and whether a fan sits on it, which usually identifies a fan curve that cannot span the pressure swings imposed on it), or if the residual has not changed by one part in 10^9 for 5 iterations while within 10× the tolerance. The latter is reported as convergence at the precision limit of double-precision arithmetic: the Newton step no longer moves the pressures. The hard cap is 2000 iterations.

Initial estimate. Free nodes start at the first boundary pressure plus the weight of air down to their elevation, $P_i^{(0)} = P_{\text{ref}} + \rho_0 g\,(z_{\text{ref}} - z_i)$, so deep nodes do not start at surface pressure. A previous solution is used instead when available (a warm start, e.g. after a small edit).

A12. Summary of one iteration

1. Refresh densities: $\rho_i(P_i, T_i, \text{moisture})$ at every node (Eq. A6); ρ_e = mean of the end nodes; $R_e = R_{\text{std}}\,\rho_e/\rho_{\text{ref}}$.
2. For every airway: Δ_e (Eq. A7), q_e (Eq. A11 or the fan root), slope s_e (Eq. A14, capped).
3. For every free node: residual F_i (Eq. A10) and Jacobian row (Eq. A15).
4. Solve $J\,\delta P = -F$ (Part B: envelope Cholesky, or preconditioned conjugate gradients for very large networks).
5. Line search on α (Eq. A16); update $P \leftarrow P + \alpha\,\delta P$.
6. Refresh densities at the new pressures; measure $\max|F|$; compare with the tolerance (Eq. A17); check stall and precision limit.

The following sections run this procedure on four test networks. All numbers come from the solver itself (an observation hook records every quantity of every iteration), and each step is shown in enough detail to reproduce by hand or in a spreadsheet.

A13. Example 1 — two parallel drives (case02)

Two identical 5 m × 5 m drives, 100 m long, connect intake A (101 425 Pa) to exhaust B (101 325 Pa). Both nodes are fixed, so there is no free node and nothing to iterate: every airway's flow follows directly from Eq. A11. The expected answer (McPherson, parallel airways) is 279.508 m³/s each at standard density.

$$\Delta = 101\,425 - 101\,325 = 100\ \text{Pa}, \qquad q = \sqrt{\frac{100}{1.2795 \times 10^{-3}}} = 279.566\ \text{m}^3/\text{s}$$

each, 559.132 m³/s in total, with mass flow $1.19951 \times 279.566 = 335.341$ kg/s per drive. The 0.02 % difference from the textbook 279.508 m³/s is the density correction: the air here is 1.1995 kg/m³, not 1.2, so R is 0.04 % lower and the flow 0.02 % higher ($279.508 \times \sqrt{1.2/1.1995} = 279.566$).

Table 2. case02: result.

Airway	R (Ns²/m⁸)	ρ (kg/m³)	q (m³/s)	ρq (kg/s)	Expected (m³/s)
t1	1.2795×10^{-3}	1.19951	279.566	335.341	279.508
t2	1.2795×10^{-3}	1.19951	279.566	335.341	279.508

A14. Example 2 — a fan on a curve (case05)

A fan with characteristic $P = 2000 - 0.2Q^2$ (tabulated every 2 m³/s from 0 to 100 m³/s) sits in airway A→M, and a system resistance $R = 0.5$ in M→B; A and B are both at 101 325 Pa. The analytic operating point is where fan pressure equals system loss: $2000 - 0.2Q^2 = 0.5Q^2$, so $Q = \sqrt{2000/0.7} = 53.452$ m³/s and $P_{\text{fan}} = 1428.6$ Pa at standard density.

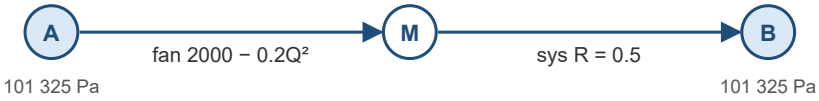


Figure 2. case05: one free node, M.

M is the only free node, so J is a 1×1 matrix and each Newton step is a division. The table below lists all eight iterations.

Table 3. case05: every Newton iteration.

Iter.	P_M before (Pa)	max |F| before (kg/s)	α taken	max |F| after (kg/s)	Tolerance (kg/s)
1	101325.0000	119.89	0.001953	119.89	1.199×10^{-4}
2	101325.0000	119.89	0.062500	119.78	1.199×10^{-4}
3	101325.0039	119.78	1.000000	114.60	1.196×10^{-4}
4	101333.7970	114.60	1.000000	75.668	1.084×10^{-4}
5	101696.2926	75.668	1.000000	6.2322	1.000×10^{-4}
6	102678.4347	6.2322	1.000000	0.065927	1.000×10^{-4}
7	102760.6218	0.065927	1.000000	3.3023×10^{-4}	1.000×10^{-4}
8	102759.7887	3.3023×10^{-4}	1.000000	1.6665×10^{-6}	1.000×10^{-4}

Iterations 1–2: the zero-flow start. M starts at 101 325 Pa, equal to B, so the system airway M→B carries no flow and its slope is capped at 10^6. The Jacobian is dominated by that cap ($J = -1.1989 \times 10^6$) and the step is a mere 10.000×10^{-5} Pa. In iteration 1 no trial passes the Armijo test, so the smallest step is taken; in iteration 2 a sixteenth of the step passes. Even these tiny pressure rises start air moving through M→B, its slope drops to a realistic value, and from iteration 3 Newton takes full steps.

Iteration 3 in full:

Starting point of iteration 3. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 4. case05, iteration 3: node pressures before the step.

Node	Role	Pressure P (Pa)
A	fixed (boundary)	101325.0000
M	free (unknown)	101325.0039
B	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 5. case05, iteration 3: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
fan	A→M	−0.0039	0.000	−0.0039	1.0000×10⁻⁶	99.99965	1.19891	119.8907	0.048834
sys	M→B	0.0039	0.000	0.0039	0.50000	0.08839	1.19891	0.1060	11.314

Airway **fan** carries a fan, so its flow is the root of $R\,q|q| - P_{\mathrm{fan}}(q) = \Delta$ on the tabulated curve: $q = 99.99965$ m³/s, where the curve gives $P_{\mathrm{fan}} = 0.014$ Pa. Check: $1.000 \times 10^{-6} \times 99.9996^2 - 0.014 = -0.004$ Pa, which equals $\Delta = -0.004$ Pa.

For example, airway **sys**: $\Delta = 101325.004 - 101325.000 - (0.000) = 0.0039$ Pa, so $q = \sqrt{0.0039/0.50000} = 0.08839$ m³/s, and $s = 1/(2 \times 0.50000 \times 0.0884) = 11.314$.

Step 2: the mass residual at every free node. $F_i = \sum_{\mathrm{in}} \rho_e q_e - \sum_{\mathrm{out}} \rho_e q_e$ (kg/s), with “in” meaning airways whose target is node i.

$$F_M = 1.19891 \times (99.9996) - 1.19891 \times (0.0884) = 119.78469 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{MM} = -(1.19891 \times 0.048834 + 1.19891 \times 11.314) = -13.6225$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\delta P_M = -\frac{F}{J} = -\frac{119.78469}{-13.6225} = 8.79312 \text{ Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 119.78$ kg/s), otherwise α is halved.

Table 6. case05, iteration 3: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	M: 101333.7970	114.60	118.59	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 114.60$ kg/s against a tolerance of 1.1963×10^{-4} kg/s, so the solver continues.

Iteration 3 is still far from the answer (the fan delivers nearly 100 m³/s into a node that lets out almost nothing) but the step is in the right direction. The residual falls 114.6 → 75.7 → 6.23 → 0.0659 → 3.3×10⁻⁴ → 1.7×10⁻⁶ kg/s: the last steps show the quadratic convergence. The final iteration in full:

Starting point of iteration 8. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 7. case05, iteration 8: node pressures before the step.

Node	Role	Pressure P (Pa)
A	fixed (boundary)	101325.0000
M	free (unknown)	102759.7887
B	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 8. case05, iteration 8: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
fan	A→M	−1434.7887	0.000	−1434.7887	1.0000×10⁻⁶	53.56816	1.20744	64.6802	0.046879
sys	M→B	1434.7887	0.000	1434.7887	0.50000	53.56844	1.20744	64.6805	0.018668

Airway **fan** carries a fan, so its flow is the root of $R\,q|q| - P_{\mathrm{fan}}(q) = \Delta$ on the tabulated curve: $q = 53.56816$ m³/s, where the curve gives $P_{\mathrm{fan}} = 1434.792$ Pa. Check: $1.000 \times 10^{-6} \times 53.5682^2 - 1434.792 = -1434.789$ Pa, which equals $\Delta = -1434.789$ Pa. On this part of the curve the fan's slope is the 52–54 m³/s segment's, −21.2 Pa per m³/s at standard density, scaled by ρ/1.2.

For example, airway **sys**: $\Delta = 102759.789 - 101325.000 - (0.000) = 1434.7887$ Pa, so $q = \sqrt{1434.7887/0.50000} = 53.56844$ m³/s, and $s = 1/(2 \times 0.50000 \times 53.5684) = 0.018668$.

Step 2: the mass residual at every free node. $F_i = \sum_{\mathrm{in}} \rho_e q_e - \sum_{\mathrm{out}} \rho_e q_e$ (kg/s), with “in” meaning airways whose target is node i.

$$F_M = 1.20744 \times (53.5682) - 1.20744 \times (53.5684) = -0.00033 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{MM} = -(1.20744 \times 0.046879 + 1.20744 \times 0.018668) = -0.0791436$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\delta P_M = -\frac{F}{J} = -\frac{-0.00033}{-0.0791436} = -4.17259 \times 10^{-3}\ \text{Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 3.3023 \times 10^{-4}$ kg/s), otherwise α is halved.

Table 9. case05, iteration 8: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	M: 102759.7846	2.9644×10^{-11}	3.2693×10^{-4}	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 1.6665 \times 10^{-6}$ kg/s against a tolerance of 1.0000×10^{-4} kg/s — **converged**.

Check against theory. At standard density the analytic operating point is 53.452 m³/s and 1428.6 Pa. In the program the air at the fan is denser, $\rho = 1.20744$ kg/m³ (the fan raises the pressure at M by about 1.4 kPa, which compresses the air), so by Eq. A9a its curve is scaled by $s = \rho/\rho_{\text{ref}} = 1.20744/1.2 = 1.00620$. The system airway's $R = 0.5$ is a stated resistance, which does not scale with density, so the balance is $s\,(2000 - 0.2Q^2) = 0.5\,Q^2$ and

$$Q = \sqrt{\frac{2000\,s}{0.2\,s + 0.5}} = \sqrt{\frac{2000 \times 1.00620}{0.2 \times 1.00620 + 0.5}} = 53.570\ \text{m}^3/\text{s}, \qquad P_{\text{fan}} = 0.5\,Q^2 = 1434.9\ \text{Pa}$$

The solver's converged values are 53.568 m³/s and 1434.8 Pa, in agreement to better than 0.01 % (the density at the fan is the one of the last iteration). (Had R been computed from the airway's geometry it would scale with density too, and the two effects would nearly cancel in $Q = \sqrt{P_0/(k + R)}$; the fixed R is what lets the flow rise by 0.2 %. With the curve used as entered, Section A6, the program gives 53.450, the textbook value.)

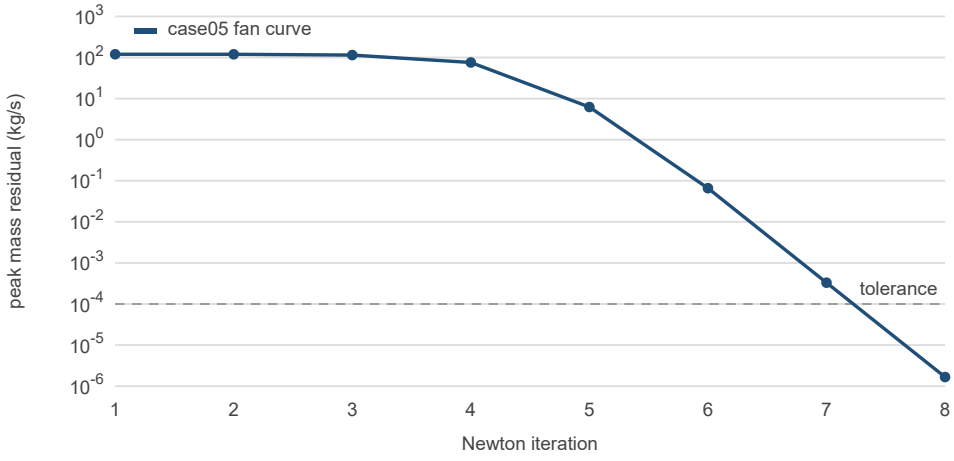


Figure 3. case05: peak mass residual after each iteration. Slow while the start-up zero flow is overcome, then quadratic.

A15. Example 3 — the balanced Wheatstone bridge (lit2)

The network of Figure 1. Resistances are chosen so that $R_{AB}/R_{BD} = R_{AC}/R_{CD} = 2/3$: the bridge is balanced, the pressures at B and C are equal, and the diagonal BC carries no air. Each side then carries $Q = \sqrt{\Delta P/(R_1 + R_2)}$: $\sqrt{300/0.5} = 24.495$ m³/s through B and $\sqrt{300/1.0} = 17.321$ m³/s through C (McPherson 1993, ch. 7). This case is a severe test of the solver: the true solution has an airway at exactly zero flow, the singular point of Section A10.

Table 10. lit2: every Newton iteration.

Iter.	P_B before (Pa)	P_C before (Pa)	max |F| before (kg/s)	α taken	max |F| after (kg/s)	Tolerance (kg/s)
1	101625.0000	101625.0000	37.969	0.001953	37.968	1.000×10^{-4}
2	101625.0000	101625.0000	37.968	0.250000	37.854	1.000×10^{-4}
3	101624.9984	101624.9984	37.854	1.000000	35.000	1.000×10^{-4}
4	101623.8647	101623.8651	35.000	1.000000	22.091	1.000×10^{-4}
5	101597.4306	101597.4303	22.091	1.000000	4.9343	1.000×10^{-4}
6	101529.0032	101529.0035	4.9343	1.000000	0.14777	1.000×10^{-4}
7	101505.7930	101505.7927	0.14777	1.000000	0.029374	1.000×10^{-4}
8	101505.2138	101505.2141	0.029374	0.500000	8.7610×10^{-4}	1.000×10^{-4}
9	101505.2138	101505.2138	8.7610×10^{-4}	1.000000	8.5898×10^{-4}	1.000×10^{-4}
10	101505.2137	101505.2137	8.5898×10^{-4}	0.500000	6.4794×10^{-6}	1.000×10^{-4}

Iteration 1. B and C start at the intake pressure (all nodes are at the same elevation), so AB, AC and BC all have zero flow and the 10^6 slope cap. The capped slopes dominate the Jacobian, and the step is microscopic:

Starting point of iteration 1. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 11. lit2, iteration 1: node pressures before the step.

Node	Role	Pressure P (Pa)
A	fixed (boundary)	101625.0000
B	free (unknown)	101625.0000
C	free (unknown)	101625.0000
D	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 12. lit2, iteration 1: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
AB	A→B	0.0000	0.000	0.0000	0.20000	0.00000	1.20248	0.0000	10^6 (cap)
AC	A→C	0.0000	0.000	0.0000	0.40000	0.00000	1.20248	0.0000	10^6 (cap)
BD	B→D	300.0000	0.000	300.0000	0.30000	31.62278	1.20069	37.9693	0.052705
CD	C→D	300.0000	0.000	300.0000	0.60000	22.36068	1.20069	26.8483	0.037268
BC	B→C	0.0000	0.000	0.0000	0.50000	0.00000	1.20248	0.0000	10^6 (cap)

For example, airway **BD**: $\Delta = 101625.000 - 101325.000 - (0.000) = 300.0000$ Pa, so $q = \sqrt{300.0000/0.30000} = 31.62278$ m³/s, and $s = 1/(2 \times 0.30000 \times 31.6228) = 0.052705$.

Step 2: the mass residual at every free node. $F_i = \sum_{\text{in}} \rho_e q_e - \sum_{\text{out}} \rho_e q_e$ (kg/s), with "in" meaning airways whose target is node i.

$$F_B = 1.20248 \times (0.0000) - 1.20069 \times (31.6228) - 1.20248 \times (0.0000) = -37.96926 \text{ kg/s}$$

$$F_C = 1.20248 \times (0.0000) - 1.20069 \times (22.3607) + 1.20248 \times (0.0000) = -26.84832 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{BB} = -(1.20248 \times 1.0000 \times 10^6 + 1.20069 \times 0.052705 + 1.20248 \times 1.0000 \times 10^6) = -2.40495 \times 1$$

$$J_{CC} = -(1.20248 \times 1.0000 \times 10^6 + 1.20069 \times 0.037268 + 1.20248 \times 1.0000 \times 10^6) = -2.40495 \times 1$$

$$J_{BC} = J_{CB} = 1.20248 \times 10^6$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\begin{bmatrix} -2.40495 \times 10^6 & 1.20248 \times 10^6 \\ 1.20248 \times 10^6 & -2.40495 \times 10^6 \end{bmatrix} \begin{bmatrix} \delta P_B \\ \delta P_C \end{bmatrix} = \begin{bmatrix} 37.9693 \\ 26.8483 \end{bmatrix}$$

By Cramer's rule, with determinant $\det J = J_{11}J_{22} - J_{12}J_{21} = 4.33785 \times 10^{12}$:

$$\delta P_B = \frac{(37.9693)(-2.40495 \times 10^6) - (1.20248 \times 10^6)(26.8483)}{4.33785 \times 10^{12}} = -2.84931 \times 10^{-5} \text{ Pa}$$

$$\delta P_C = \frac{(-2.40495 \times 10^6)(26.8483) - (1.20248 \times 10^6)(37.9693)}{4.33785 \times 10^{12}} = -2.54103 \times 10^{-5} \text{ Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 37.969$ kg/s), otherwise α is halved.

Table 13. lit2, iteration 1: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	B: 101625.0000 C: 101625.0000	37.952	37.590	rejected, halve α
2	0.500000	B: 101625.0000 C: 101625.0000	37.957	37.779	rejected, halve α
3	0.250000	B: 101625.0000 C: 101625.0000	37.961	37.874	rejected, halve α
4	0.125000	B: 101625.0000 C: 101625.0000	37.963	37.922	rejected, halve α
5	0.062500	B: 101625.0000 C: 101625.0000	37.965	37.946	rejected, halve α
6	0.031250	B: 101625.0000 C: 101625.0000	37.966	37.957	rejected, halve α
7	0.015625	B: 101625.0000 C: 101625.0000	37.967	37.963	rejected, halve α
8	0.007813	B: 101625.0000 C: 101625.0000	37.968	37.966	rejected, halve α
9	0.003906	B: 101625.0000 C: 101625.0000	37.968	37.968	rejected, halve α
10	0.001953	B: 101625.0000 C: 101625.0000	37.968	37.969	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 37.968$ kg/s against a tolerance of 1.0000×10^{-4} kg/s, so the solver continues.

Iterations 2–7. The slopes fall as flows develop (iteration 2: 4740 and 3548; iteration 3: 27.7 and 19.7), the Armijo test passes at α = 1 from iteration 3, and the residual drops 35.0 → 22.1 → 4.93 → 0.148 → 0.0294 kg/s. Iteration 5 in full:

Starting point of iteration 5. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 14. lit2, iteration 5: node pressures before the step.

Node	Role	Pressure P (Pa)
A	fixed (boundary)	101625.0000
B	free (unknown)	101597.4306
C	free (unknown)	101597.4303
D	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 15. lit2, iteration 5: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
AB	A→B	27.5694	0.000	27.5694	0.20000	11.74083	1.20231	14.1161	0.21293
AC	A→C	27.5697	0.000	27.5697	0.40000	8.30207	1.20231	9.9817	0.15056
BD	B→D	272.4306	0.000	272.4306	0.30000	30.13473	1.20053	36.1776	0.055307
CD	C→D	272.4303	0.000	272.4303	0.60000	21.30846	1.20053	25.5814	0.039108
BC	B→C	0.0003	0.000	0.0003	0.50000	0.02452	1.20215	0.0295	40.788

For example, airway **AB**: $\Delta = 101625.000 - 101597.431 - (0.000) = 27.5694$ Pa, so $q = \sqrt{27.5694/0.20000} = 11.74083$ m³/s, and $s = 1/(2 \times 0.20000 \times 11.7408) = 0.21293$.

Step 2: the mass residual at every free node. $F_i = \sum_{\text{in}} \rho_e q_e - \sum_{\text{out}} \rho_e q_e$ (kg/s), with "in" meaning airways whose target is node i.

$$F_B = 1.20231 \times (11.7408) - 1.20053 \times (30.1347) - 1.20215 \times (0.0245) = -22.09096 \text{ kg/s}$$

$$F_C = 1.20231 \times (8.3021) - 1.20053 \times (21.3085) + 1.20215 \times (0.0245) = -15.57029 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{BB} = -(1.20231 \times 0.21293 + 1.20053 \times 0.055307 + 1.20215 \times 40.788) = -49.3556$$

$$J_{CC} = -(1.20231 \times 0.15056 + 1.20053 \times 0.039108 + 1.20215 \times 40.788) = -49.2612$$

$$J_{BC} = J_{CB} = 49.0332$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\begin{bmatrix} -49.3556 & 49.0332 \\ 49.0332 & -49.2612 \end{bmatrix} \begin{bmatrix} \delta P_B \\ \delta P_C \end{bmatrix} = \begin{bmatrix} 22.0910 \\ 15.5703 \end{bmatrix}$$

By Cramer's rule, with determinant $\det J = J_{11}J_{22} - J_{12}J_{21} = 27.0607$:

$$\delta P_B = \frac{(22.0910)(-49.2612) - (49.0332)(15.5703)}{27.0607} = -68.4273 \text{ Pa}$$

$$\delta P_C = \frac{(-49.3556)(15.5703) - (49.0332)(22.0910)}{27.0607} = -68.4267 \text{ Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 22.091$ kg/s), otherwise α is halved.

Table 16. lit2, iteration 5: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	B: 101529.0032 C: 101529.0035	4.9359	21.870	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 4.9343$ kg/s against a tolerance of 1.0000×10^{-4} kg/s, so the solver continues.

Iterations 8–10: the diagonal near zero. Now the flow in BC is tiny (≈ 10^{-3} m³/s), its slope large (≈ 1400), and Section A10's mirror effect appears: the full step would flip the sign of the BC flow. In iteration 8 the line search rejects the full step and halves it; iteration 9 takes a full step that barely helps ($8.76 \times 10^{-4} \rightarrow 8.59 \times 10^{-4}$), and iteration 10 halves again and lands at 6.5×10^{-6} kg/s, below the 10^{-4} kg/s tolerance. The last iteration in full:

Starting point of iteration 10. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 17. lit2, iteration 10: node pressures before the step.

Node	Role	Pressure P (Pa)
A	fixed (boundary)	101625.0000
B	free (unknown)	101505.2137
C	free (unknown)	101505.2137
D	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 18. lit2, iteration 10: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
AB	A→B	119.7863	0.000	119.7863	0.20000	24.47308	1.20176	29.4109	0.10215
AC	A→C	119.7863	0.000	119.7863	0.40000	17.30508	1.20176	20.7966	0.072233
BD	B→D	180.2137	0.000	180.2137	0.30000	24.50943	1.19998	29.4109	0.068001
CD	C→D	180.2137	0.000	180.2137	0.60000	17.33079	1.19998	20.7966	0.048084
BC	B→C	2.56×10^{-7}	0.000	2.56×10^{-7}	0.50000	0.00072	1.20105	0.0009	1398.3

For example, airway **AB**: $\Delta = 101625.000 - 101505.214 - (0.000) = 119.7863$ Pa, so $q = \sqrt{119.7863/0.20000} = 24.47308$ m³/s, and $s = 1/(2 \times 0.20000 \times 24.4731) = 0.10215$.

Step 2: the mass residual at every free node. $F_i = \sum_{\text{in}} \rho_e q_e - \sum_{\text{out}} \rho_e q_e$ (kg/s), with "in" meaning airways whose target is node i.

$$F_B = 1.20176 \times (24.4731) - 1.19998 \times (24.5094) - 1.20105 \times (0.0007) = -0.00086 \text{ kg/s}$$

$$F_C = 1.20176 \times (17.3051) - 1.19998 \times (17.3308) + 1.20105 \times (0.0007) = 0.00086 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{BB} = -(1.20176 \times 0.10215 + 1.19998 \times 0.068001 + 1.20105 \times 1398.3) = -1679.59$$

$$J_{CC} = -(1.20176 \times 0.072233 + 1.19998 \times 0.048084 + 1.20105 \times 1398.3) = -1679.53$$

$$J_{BC} = J_{CB} = 1679.39$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\begin{bmatrix} -1679.59 & 1679.39 \\ 1679.39 & -1679.53 \end{bmatrix} \begin{bmatrix} \delta P_B \\ \delta P_C \end{bmatrix} = \begin{bmatrix} 8.58981 \times 10^{-4} \\ -8.58981 \times 10^{-4} \end{bmatrix}$$

By Cramer's rule, with determinant $\det J = J_{11}J_{22} - J_{12}J_{21} = 585.920$:

$$\delta P_B = \frac{(8.58981 \times 10^{-4})(-1679.53) - (1679.39)(-8.58981 \times 10^{-4})}{585.920} = -2.11737 \times 10^{-7}\ \text{Pa}$$

$$\delta P_C = \frac{(-1679.59)(-8.58981 \times 10^{-4}) - (1679.39)(8.58981 \times 10^{-4})}{585.920} = 2.99721 \times 10^{-7}\ \text{Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 8.5898 \times 10^{-4}$ kg/s), otherwise α is halved.

Table 19. lit2, iteration 10: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	B: 101505.2137 C: 101505.2137	8.5896×10⁻⁴	8.5039×10⁻⁴	rejected, halve α
2	0.500000	B: 101505.2137 C: 101505.2137	6.4794×10⁻⁶	8.5469×10⁻⁴	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 6.4794 \times 10^{-6}$ kg/s against a tolerance of 1.0000×10^{-4} kg/s — **converged**.

Check against theory. Flows $q_{AB} = 24.473$, $q_{BD} = 24.509$ (24.495 expected); $q_{AC} = 17.305$, $q_{CD} = 17.331$ (17.321 expected); $q_{BC} = 0.00072$ (0 expected). B and C are at 101 505.214 Pa, both $101\,625 - 0.2 \times 24.495^2 = 101\,505.0$ Pa by hand at standard density. The differences of a few hundredths of a m³/s between AB and BD are the density correction: air expands slightly from A to D (1.2018 → 1.2000 kg/m³), so the same mass is a larger volume downstream.

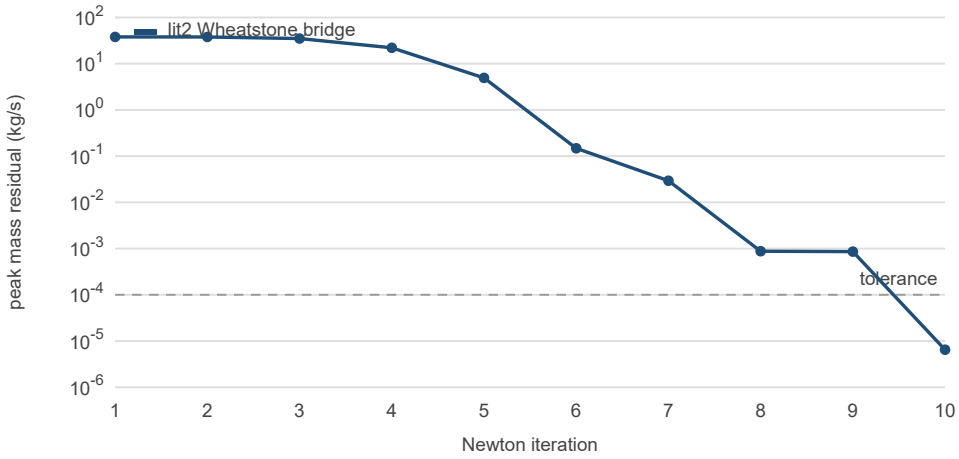


Figure 4. lit2: peak mass residual after each iteration.

A16. Example 4 — natural ventilation (case09)

A 500 m downcast shaft S1→B1 at 5 °C, a 400 m level B1→B2 and a 500 m upcast B2→S2 at 35 °C. Both shaft collars are at 101 325 Pa and there is no fan: only the density difference between the shafts drives air. B1 and B2 are free.

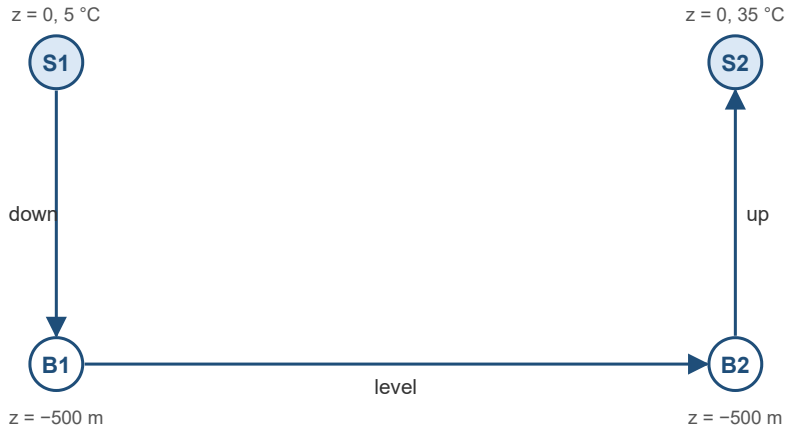


Figure 5. case09: natural ventilation through two shafts.

Initial estimate. The free nodes start at the collar pressure plus the weight of a 500 m column at the collar's density: $101\,325 + \rho_0 \times 9.79 \times 500 = 107527.1$ Pa, so $\rho_0 = 1.26703$ kg/m³ (5 °C air at 101 325 Pa, 50 % RH).

The hydrostatic terms. In the first iteration the downcast density is 1.30587 kg/m³ and the upcast 1.16860; the column terms are $\rho g\,\Delta z = 1.30587 \times 9.79 \times (-500) = -6392.3$ Pa down and $1.16860 \times 9.79 \times (+500) = 5720.3$ Pa up. Around the loop they leave 672.0 Pa uncancelled. That is the natural ventilation pressure, which the friction of the three airways must absorb.

Table 20. case09: every Newton iteration.

Iter.	P_{B1} before (Pa)	P_{B2} before (Pa)	max |F| before (kg/s)	α taken	max |F| after (kg/s)	Tolerance (kg/s)
1	107527.1312	107527.1312	573.06	1.000000	497.96	4.983×10^{-4}
2	107406.4455	107406.4451	497.96	1.000000	473.57	4.911×10^{-4}
3	107396.9629	107395.7280	473.57	1.000000	341.05	4.679×10^{-4}
4	107426.6222	107361.9325	341.05	1.000000	83.398	3.630×10^{-4}
5	107544.7530	107230.1341	83.398	1.000000	0.94867	3.175×10^{-4}
6	107586.7929	107183.7470	0.94867	1.000000	0.018189	3.169×10^{-4}
7	107587.2888	107183.3424	0.018189	1.000000	3.7709×10^{-4}	3.169×10^{-4}
8	107587.2990	107183.3372	3.7709×10^{-4}	1.000000	8.3963×10^{-6}	3.169×10^{-4}

Iteration 1 in full (note the 10^6 cap on the level, which starts with B1 = B2):

Starting point of iteration 1. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 21. case09, iteration 1: node pressures before the step.

Node	Role	Pressure P (Pa)
S1	fixed (boundary)	101325.0000
B1	free (unknown)	107527.1312
B2	free (unknown)	107527.1312
S2	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial\Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 22. case09, iteration 1: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
down	S1→B1	−6202.1312	−6392.258	190.1263	2.2392×10^{-3}	291.39296	1.30587	380.5227	0.76631
level	B1→B2	0.0000	0.000	0.0000	6.5238×10^{-3}	0.00000	1.27419	0.0000	10^6 (cap)
up	B2→S2	6202.1312	5720.276	481.8551	2.0038×10^{-3}	490.38236	1.16860	573.0587	0.50885

For example, airway **down**: $\Delta = 101325.000 - 107527.131 - (-6392.258) = 190.1263$ Pa, so $q = \sqrt{190.1263/2.2392\times10^{-3}} = 291.39296$ m³/s, and $s = 1/(2\times2.2392\times10^{-3}\times291.3930) = 0.76631$.

Step 2: the mass residual at every free node. $F_i = \sum_{\text{in}}\rho_e q_e - \sum_{\text{out}}\rho_e q_e$ (kg/s), with "in" meaning airways whose target is node i.

$$F_{B1} = 1.30587\times(291.3930) - 1.27419\times(0.0000) = 380.52275\ \text{kg/s}$$

$$F_{B2} = 1.27419\times(0.0000) - 1.16860\times(490.3824) = -573.05873\ \text{kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{B1B1} = -(1.30587 \times 0.76631 + 1.27419 \times 1.0000 \times 10^6) = -1.27419 \times 10^6$$

$$J_{B2B2} = -(1.27419 \times 1.0000 \times 10^6 + 1.16860 \times 0.50885) = -1.27419 \times 10^6$$

$$J_{B1B2} = J_{B2B1} = 1.27419 \times 10^6$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\begin{bmatrix} -1.27419 \times 10^6 & 1.27419 \times 10^6 \\ 1.27419 \times 10^6 & -1.27419 \times 10^6 \end{bmatrix} \begin{bmatrix} \delta P_{B1} \\ \delta P_{B2} \end{bmatrix} = \begin{bmatrix} -380.523 \\ 573.059 \end{bmatrix}$$

By Cramer's rule, with determinant $\det J = J_{11}J_{22} - J_{12}J_{21} = 2.03277 \times 10^6$:

$$\delta P_{B1} = \frac{(-380.523)(-1.27419 \times 10^6) - (1.27419 \times 10^6)(573.059)}{2.03277 \times 10^6} = -120.686 \text{ Pa}$$

$$\delta P_{B2} = \frac{(-1.27419 \times 10^6)(573.059) - (1.27419 \times 10^6)(-380.523)}{2.03277 \times 10^6} = -120.686 \text{ Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 573.06$ kg/s), otherwise α is halved.

Table 23. case09, iteration 1: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	B1: 107406.4455 B2: 107406.4451	495.82	567.33	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 497.96$ kg/s against a tolerance of 4.9827×10^{-4} kg/s, so the solver continues.

From iteration 2 every step is a full Newton step, and the residual falls 474 → 341 → 83.4 → 0.949 → 0.0182 → 3.8 × 10⁻⁴ → 8.4 × 10⁻⁶ kg/s. The last iteration in full:

Starting point of iteration 8. The node pressures entering this iteration are listed below; the boundary pressures never change.

Table 24. case09, iteration 8: node pressures before the step.

Node	Role	Pressure P (Pa)
S1	fixed (boundary)	101325.0000
B1	free (unknown)	107587.2990
B2	free (unknown)	107183.3372
S2	fixed (boundary)	101325.0000

Step 1: the flow in every airway. For each airway the effective pressure difference is $\Delta = P_s - P_t - \rho g\,(z_t - z_s)$, and the flow follows from the inverted square law $q = \mathrm{sign}(\Delta)\sqrt{|\Delta|/R}$. The slope $s = \partial q/\partial \Delta = 1/(2R|q|)$ is what the Jacobian needs.

Table 25. case09, iteration 8: airway flows and slopes.

Airway	s→t	P_s − P_t (Pa)	ρgΔz (Pa)	Δ (Pa)	R (Ns²/m⁸)	q (m³/s)	ρ (kg/m³)	ρq (kg/s)	s (m³/s per Pa)
down	S1→B1	−6262.2990	−6394.102	131.8030	2.2398×10⁻³	242.58167	1.30625	316.8727	0.92024
level	B1→B2	403.9617	0.000	403.9617	6.5158×10⁻³	248.99231	1.27262	316.8723	0.30819
up	B2→S2	5858.3372	5710.763	147.5741	2.0004×10⁻³	271.60832	1.16665	316.8725	0.92024

For example, airway **down**: $\Delta = 101325.000 - 107587.299 - (-6394.102) = 131.8030$ Pa, so $q = \sqrt{131.8030/2.2398 \times 10^{-3}} = 242.58167$ m³/s, and $s = 1/(2 \times 2.2398 \times 10^{-3} \times 242.5817) = 0.92024$.

Step 2: the mass residual at every free node. $F_i = \sum_{\text{in}} \rho_e q_e - \sum_{\text{out}} \rho_e q_e$ (kg/s), with "in" meaning airways whose target is node i.

$$F_{B1} = 1.30625 \times (242.5817) - 1.27262 \times (248.9923) = 0.00038 \text{ kg/s}$$

$$F_{B2} = 1.27262 \times (248.9923) - 1.16665 \times (271.6083) = -0.00014 \text{ kg/s}$$

Step 3: the Jacobian. Raising P_i by 1 Pa raises Δ on every airway leaving i and lowers it on every airway entering i, so $J_{ii} = -\sum_e \rho_e s_e$ over the airways at i, and $J_{ij} = +\rho_e s_e$ for an airway between i and j.

$$J_{B1B1} = -(1.30625 \times 0.92024 + 1.27262 \times 0.30819) = -1.59428$$

$$J_{B2B2} = -(1.27262 \times 0.30819 + 1.16665 \times 0.92024) = -1.46581$$

$$J_{B1B2} = J_{B2B1} = 0.392206$$

Step 4: the Newton step. Solve $J\,\delta P = -F$.

$$\begin{bmatrix} -1.59428 & 0.392206 \\ 0.392206 & -1.46581 \end{bmatrix} \begin{bmatrix} \delta P_{B1} \\ \delta P_{B2} \end{bmatrix} = \begin{bmatrix} -3.77091 \times 10^{-4} \\ 1.44348 \times 10^{-4} \end{bmatrix}$$

By Cramer's rule, with determinant $\det J = J_{11}J_{22} - J_{12}J_{21} = 2.18308$:

$$\delta P_{B1} = \frac{(-3.77091 \times 10^{-4})(-1.46581) - (0.392206)(1.44348 \times 10^{-4})}{2.18308} = 2.27261 \times 10^{-4} \text{ Pa}$$

$$\delta P_{B2} = \frac{(-1.59428)(1.44348 \times 10^{-4}) - (0.392206)(-3.77091 \times 10^{-4})}{2.18308} = -3.76682 \times 10^{-5} \text{ Pa}$$

Step 5: the line search. The full step is tried first. It is accepted if the largest nodal residual falls to at most $(1 - 0.01\,\alpha)$ times its value before the step ($\max|F| = 3.7709 \times 10^{-4}$ kg/s), otherwise α is halved.

Table 26. case09, iteration 8: line search.

Trial	α	P after trial	max |F| after trial (kg/s)	Required ≤ (kg/s)	Result
1	1.000000	B1: 107587.2992 B2: 107183.3372	9.3053×10^{-11}	3.7332×10^{-4}	accepted

After the step the densities are refreshed at the new pressures and the residual is measured again: $\max|F| = 8.3963\times10^{-6}$ kg/s against a tolerance of 3.1687×10^{-4} kg/s — **converged**.

Check: the loop balance. The NVP at the converged state is 683.3 Pa. The friction losses are Rq^2: $2.2398\times10^{-3}\times242.58^2 + 6.5158\times10^{-3}\times248.99^2 + 2.0004\times10^{-3}\times271.61^2 = 683.3$ Pa. The two agree, as Kirchhoff's second law requires. The volume flows differ (242.6, 249.0, 271.6 m³/s) while the mass flow is the same in all three (316.87, 316.87, 316.87 kg/s): the warm upcast air is lighter, so the same mass takes more volume. The solver reports a 17 % density spread here and recommends a thermal solve. In this test the temperatures are imposed at the nodes; Part C computes them instead.

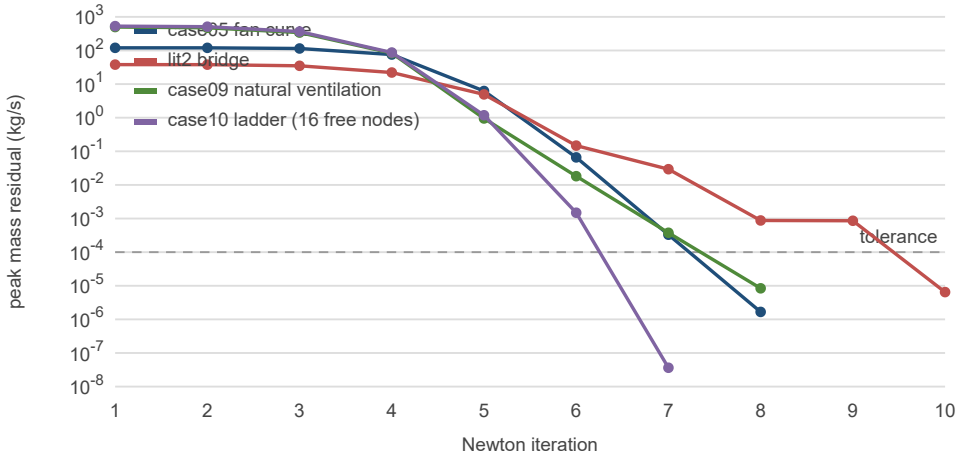


Figure 6. Residual histories of all four examples and the 16-node ladder.

A17. Larger networks

Nothing changes for a large mine except the size of the linear system in step 4. The ladder test case10 (two parallel drives joined by 8 crosscuts, 16 free nodes) converges in 7 iterations: 530 → 510 → 365 → 86.8 → 1.18 → 1.49×10^{-3} → 3.64×10^{-8} kg/s. The Palabora project (4,202 nodes, 4,477 airways; 2,770 free nodes after dead ends are pruned) converges in 39 iterations, each solving a 2,770-unknown system by the method of Part B. Its history (last figure of this part) shows both regimes: large full or half steps bring the residual from 1400 kg/s to about 0.01 kg/s in 19 iterations, after which the line search keeps cutting the step (down to 0.03125) and the residual falls only linearly to the tolerance (8.65×10^{-4} kg/s). That slow tail is the near-zero-flow effect of Section A10, spread over the many crosscuts and old workings of a real mine that carry almost no air. The work per iteration is dominated by the linear solve; the number of iterations depends mostly on how far the start is from the solution and on near-zero-flow airways, hardly at all on network size.

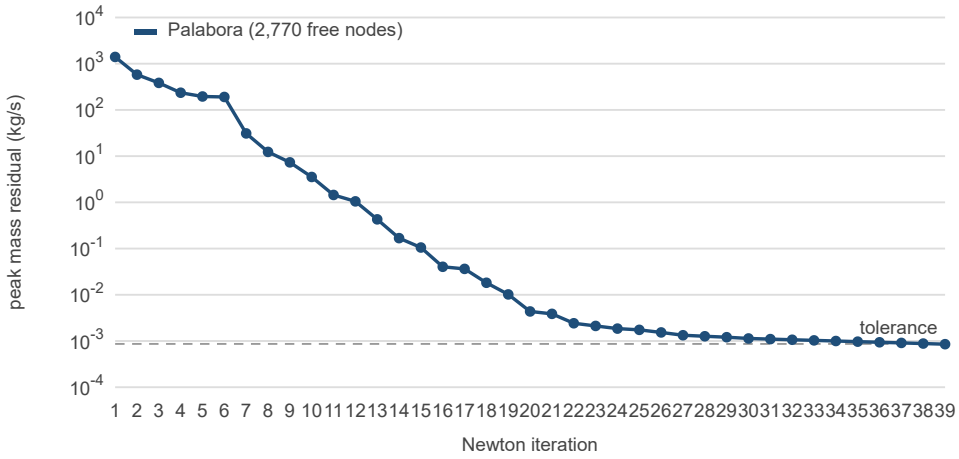


Figure 7. Palabora: peak mass residual per iteration.

Part B — Solving the linear systems

Every Newton iteration of Part A, and of the all-at-once formulation of Section C8, ends in a linear system $A\,x = b$. For a large mine this is where nearly all the computing time goes, so the choice of method matters. This part explains the methods the program uses: the **Cholesky factorisation** (direct, for the symmetric airflow system), stored in an **envelope** and made small by **reverse Cuthill–McKee ordering**; the **conjugate gradient** method with an **incomplete Cholesky** preconditioner as a fall-back for very large systems; and, for the non-symmetric coupled system, an **LU factorisation** polished by **GMRES**. The program's default coupling of airflow and heat (Section C5) solves the symmetric airflow system of Part A in every round and needs only the first of these; the non-symmetric all-at-once system of Section C8 needs the last, and is described here because it is still in the engine.

B1. The airflow system

The Newton step solves $J\,\delta P = -F$. Section A9 showed that J is symmetric with negative diagonal and nonnegative off-diagonals, weakly diagonally dominant, and strictly so in any row next to a boundary. The program multiplies by −1 and solves

$$B\,x = c, \qquad B = -J, \quad c = F, \quad x = \delta P \tag{B1}$$

B is then a **symmetric positive definite (SPD) M-matrix**: $x^{\mathrm{T}}Bx > 0$ for every $x \neq 0$. (Physically: raising a pattern of node pressures always pushes net air out of the region raised.) For SPD matrices the Cholesky factorisation exists, needs no pivoting, and is numerically stable.

B2. Gaussian elimination and the LU factorisation

Gaussian elimination subtracts multiples of row k from the rows below it to clear column k, for $k = 1, \dots, n-1$. Recording the multipliers gives the factorisation

$$A = L\,U \tag{B2}$$

with L lower triangular (unit diagonal, the multipliers $l_{ik} = a_{ik}^{(k)}/a_{kk}^{(k)}$ below it) and U upper triangular (the eliminated matrix). Then $Ax = b$ is solved in two triangular sweeps: *forward substitution* $Ly = b$, then *back substitution* $Ux = y$. Factorisation costs about $\frac{2}{3}n^3$ operations for a dense matrix, each solve $2n^2$. A zero or tiny pivot $a_{kk}^{(k)}$ breaks the method; general matrices need row exchanges (pivoting). SPD matrices never do.

B3. The Cholesky factorisation

For SPD B the factorisation can be made symmetric: $B = LL^{\mathrm{T}}$ with L lower triangular with a positive diagonal. Comparing entries of LL^{T} with B column by column gives the algorithm directly:

$$l_{jj} = \sqrt{b_{jj} - \sum_{k<j} l_{jk}^2}, \qquad l_{ij} = \frac{1}{l_{jj}}\left(b_{ij} - \sum_{k<j} l_{ik}\, l_{jk}\right) \quad (i > j) \tag{B3}$$

The square root's argument is positive at every step exactly when B is SPD, so the factorisation also *tests* definiteness: if a nonpositive value appears, the program abandons the direct solve and falls back to the iterative method of Section B6. Cholesky needs half the work and half the storage of LU ($\frac{1}{3}n^3$ operations dense) and no pivoting. The solve is then

$$L\,y = c \quad \text{(forward)}, \qquad L^{\mathrm{T}} x = y \quad \text{(backward)} \tag{B4}$$

Worked example: the bridge, iteration 5

From Section A15, iteration 5 of the bridge has (free nodes B, C)

$$B = -J = \begin{bmatrix} 49.3556 & -49.0332 \\ -49.0332 & 49.2612 \end{bmatrix}, \qquad c = F = \begin{bmatrix} -22.0910 \\ -15.5703 \end{bmatrix}$$

Column 1:

$$l_{11} = \sqrt{49.3556} = 7.02536, \qquad l_{21} = \frac{-49.0332}{7.02536} = -6.97946$$

Column 2:

$$l_{22} = \sqrt{49.2612 - (-6.97946)^2} = \sqrt{0.548279} = 0.740459$$

Forward substitution $Ly = c$:

$$y_1 = \frac{-22.0910}{7.02536} = -3.14446, \qquad y_2 = \frac{-15.5703 - (-6.97946)(-3.14446)}{0.740459} = -50.6672$$

Back substitution $L^{\mathrm{T}} x = y$:

$$x_2 = \frac{-50.6672}{0.740459} = -68.4267, \qquad x_1 = \frac{-3.14446 - (-6.97946)(-68.4267)}{7.02536} = -68.4273$$

So $\delta P_B = -68.4273$ Pa and $\delta P_C = -68.4267$ Pa, identical to the Cramer's-rule result of Section A15 and to the solver's own step (-68.4273, -68.4267).

A larger example: the 16-node ladder

The ladder (case10) has 16 free nodes: T1…T8 along one drive and B1…B8 along the other, joined by crosscuts. Each row of B has at most 4 nonzeros. The table below shows the top-left corner of B in iteration 4 (in the solver's node order T1, B1, T2, B2, …).

Table 27. case10, iteration 4: top-left 5×5 corner of B = −J (kg/s per Pa).

	T1	B1	T2	B2	T3
T1	52.513	−16.316	−25.452	0	0
B1	−16.316	44.543	0	−25.452	0
T2	−25.452	0	124.41	−38.657	−60.300
B2	0	−25.452	−38.657	124.41	0
T3	0	0	−60.300	0	294.81

The first two columns of L by Eq. B3:

$$l_{11} = \sqrt{52.513} = 7.2466, \quad l_{21} = \frac{-16.316}{7.2466} = -2.2515, \quad l_{31} = \frac{-25.452}{7.2466} = -3.5123$$

$$l_{22} = \sqrt{44.543 - (-2.2515)^2} = 6.2828, \quad l_{32} = \frac{0 - (-3.5123)(-2.2515)}{6.2828} = -1.2587$$

Notice l_{32}: b_{32} is zero (T2 and B1 are not connected), yet l_{32} is not. Entries of L that are nonzero where B is zero are called **fill-in**. In this factor there are 7 of them (out of 120 positions below the diagonal). Continuing through all 16 columns and solving gives the Newton step to within 1.68×10^{-11} Pa of the solver's own.

B4. Sparsity, fill-in and the envelope

A mine network's matrix is extremely sparse: in the Palabora model 2,770 free nodes give 8,784 nonzeros, 3.17 per row, against 7,672,900 entries in the full matrix. A dense factorisation would ignore this and need about $\frac{1}{3}n^3 \approx 7.1 \times 10^9$ operations. But the factor of a sparse matrix is not as sparse as the matrix: elimination creates fill-in. The trick is to keep the fill-in close to the diagonal.

Envelope (profile, skyline) storage. For row i let f_i be the column of its first nonzero. A key property of Cholesky (and of LU without pivoting) is that L has no nonzeros to the left of f_i in row i: fill-in appears only *inside* the envelope $\{(i,j) : f_i \le j \le i\}$. The program therefore stores each row of L from f_i to the diagonal as one contiguous array and factorises inside it, skipping everything outside. The cost depends on the envelope size, not on n^2.

Ordering. The envelope depends entirely on how the nodes are numbered: renumbering the unknowns (a symmetric permutation PBP^{T}) changes nothing about the solution but everything about the envelope. The program uses the **reverse Cuthill–McKee** (RCM) ordering:

1. Start from a node of lowest degree (fewest connections), at the periphery of the network.
2. Number its neighbours next, in order of increasing degree; then the neighbours of those, and so on: a breadth-first sweep, level by level, across the network.
3. Reverse the final order (the reversal is known to reduce fill-in further, George 1971).

Neighbouring nodes then get nearby numbers, so every row's nonzeros sit in a narrow band left of the diagonal. A mine network is close to planar (levels stacked by shafts and ramps), so the breadth-first levels stay small and the band stays narrow.

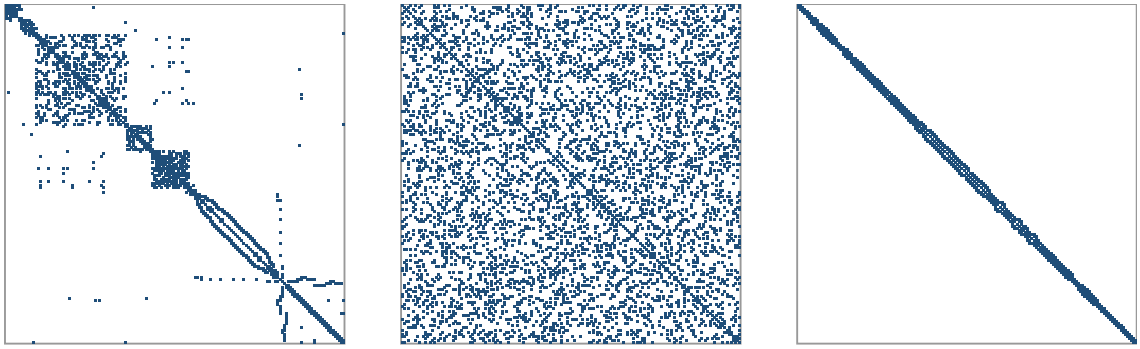


Figure 8. Nonzero pattern of the Palabora airflow matrix (2,770 × 2,770, each dot one block of about 17 × 17 entries) in three orders. Same matrix, same solution.

Table 28. Palabora: envelope size by node order.

Order	Bandwidth	Envelope (L + U entries)	Fraction of the full matrix
File order (as drawn)	2,520	375,298	4.89 %
Random order	2,758	3,979,560	51.87 %
Reverse Cuthill–McKee	42	136,378	1.78 %

RCM shrinks the envelope to 36 % of the file order and 3.4 % of a random order. The factorisation work grows roughly with the sum of the squared row lengths inside the envelope, so the gain in time is larger still. The program uses the direct solve whenever the envelope fits in 30 million entries (about 240 MB); beyond that it switches to the iterative method below.

B5. Why a direct solve

Earlier versions of the solver used the conjugate gradient method (next section) for every step. On real mine networks it struggled: airway conductances ρs range from about 4 to over 2×10^6 in the same matrix (a short wide drive next to a long narrow raise, or a near-zero-flow airway at the slope cap), and at that spread conjugate gradients lose accuracy to round-off. Measured on a 1,734-node model, 58 of 78 linear solves hit the 3,000-iteration cap with a true relative residual up to 13.5, worse than not solving at all. A direct factorisation has no such problem: its accuracy does not depend on convergence, and RCM makes it cheap.

B6. Conjugate gradients with incomplete Cholesky (fall-back)

For a network too large for the envelope, the program solves $Bx = c$ iteratively by the **preconditioned conjugate gradient** (PCG) method. CG minimises the energy norm of the error $\|x - x^*\|_B$ over a growing subspace, one search direction per iteration, with each new direction B-conjugate ($p_i^{\mathrm{T}} B p_j = 0$) to all previous ones. One iteration is

$$\alpha_k = \frac{r_k^{\mathrm{T}} z_k}{p_k^{\mathrm{T}} B p_k}, \qquad x_{k+1} = x_k + \alpha_k p_k, \qquad r_{k+1} = r_k - \alpha_k B p_k,$$
$$z_{k+1} = M^{-1} r_{k+1}, \quad p_{k+1} = z_{k+1} + \beta_k p_k, \qquad \beta_k = \frac{r_{k+1}^{\mathrm{T}} z_{k+1}}{r_k^{\mathrm{T}} z_k} \tag{B5}$$

starting from $x_0 = 0,\ r_0 = c,\ z_0 = p_0 = M^{-1}c$. In exact arithmetic CG finishes in at most n iterations; in practice the iteration count depends on the condition number of $M^{-1}B$.

Worked example (no preconditioner, $M = I$). The same bridge system as Section B3. Iteration 1: $p_0 = c = \begin{bmatrix} -22.091 \\ -15.570 \end{bmatrix}$, $Bp_0 = \begin{bmatrix} -326.85 \\ 316.18 \end{bmatrix}$, so

$$\alpha_0 = \frac{c^{\mathrm{T}} c}{p_0^{\mathrm{T}} B p_0} = 0.317935, \quad x_1 = \alpha_0 p_0 = \begin{bmatrix} -7.02350 \\ -4.95034 \end{bmatrix}, \quad r_1 = \begin{bmatrix} 81.827 \\ -116.10 \end{bmatrix}, \quad \beta_0 = 27.618$$

Iteration 2: $p_1 = r_1 + \beta_0 p_0 = \begin{bmatrix} -528.29 \\ -546.12 \end{bmatrix}$, $\alpha_1 = 0.116231$, giving

$$x_2 = \begin{bmatrix} -68.4273 \\ -68.4267 \end{bmatrix}, \qquad r_2 = \begin{bmatrix} -4.97 \times 10^{-13} \\ 4.69 \times 10^{-13} \end{bmatrix}$$

the exact solution of Section B3 after $n = 2$ iterations, as the theory says. With thousands of unknowns, the point of a preconditioner is to get there in far fewer than n iterations.

Incomplete Cholesky IC(0). The preconditioner $M = \tilde{L}\tilde{L}^{\mathrm{T}}$ is a Cholesky factorisation that keeps only the entries where B itself is nonzero, discarding all fill-in. It is as sparse as B, cheap to apply (two triangular sweeps), and close enough to B to cut the iteration count many-fold. For an M-matrix IC(0) is guaranteed to exist (Meijerink & van der Vorst 1977); for degenerate user networks the program adds a small diagonal shift if needed. RCM ordering helps here too, because IC(0)'s quality depends on the order. The PCG stops at a relative residual of 10^{-6} or $\min(3000, 3n + 50)$ iterations.

B7. The coupled system is not symmetric

In the all-at-once coupled airflow + thermal formulation (Section C8; the program's default coupling, Section C5, does not form this system) the unknowns are, at every node, the pressure P, the enthalpy h and the moisture content r. The Jacobian couples them in one direction much more strongly than the other. The heat equation at a node depends on the enthalpy upstream, but not on the enthalpy downstream: air carries heat with it, it does not carry it back. The mass equation depends on enthalpy through density, but the heat equation depends on pressure only weakly. The result is a matrix with $J_{ij} \not\equiv J_{ji}$. For example, from the first coupled iteration of case09 (Part C), at node B2:

$$\frac{\partial F_{B2}^{\mathrm{mass}}}{\partial h_{B2}} = -8.042 \ \frac{\mathrm{kg/s}}{\mathrm{kJ/kg}}, \qquad \frac{\partial F_{B2}^{\mathrm{heat}}}{\partial P_{B2}} = -1.613 \times 10^{-4} \ \frac{\mathrm{kJ/kg}}{\mathrm{Pa}}$$

CG requires a symmetric positive definite matrix, and Cholesky requires symmetry. Neither applies. The coupled system needs a method for general matrices.

B8. Why GMRES

The choice for a general sparse system is between a direct LU factorisation and a Krylov iterative method for non-symmetric matrices: GMRES, BiCGSTAB, QMR. The program combines the two:

1. **Envelope LU without pivoting**, in RCM order with each node's P, h and r numbered together. It reuses the envelope machinery of Section B4. Without pivoting the factorisation can meet a tiny pivot (a node no air reaches has a nearly empty mass row), so a pivot below 10^{-12} of its row's size is *nudged* to that size and counted. The factor is then the exact LU of a slightly different matrix.
2. **GMRES, preconditioned by that LU**, which removes the error the nudges introduced and certifies the accuracy of the answer by computing the true residual $\|b - Ax\|$.

Why GMRES rather than another Krylov method:

- **It minimises the residual.** At iteration k GMRES returns the vector in the Krylov space $\mathrm{K}_k = \mathrm{span}\{b, Ab, \ldots, A^{k-1}b\}$ with the smallest $\|b - Ax\|_2$. The residual can never increase from one iteration to the next. BiCGSTAB and CGS do not minimise anything and can stagnate or break down on exactly the badly scaled, strongly non-symmetric matrices the coupled problem produces (mass rows in kg/s next to enthalpy rows in kJ/kg).
- **It works for any nonsingular matrix**, with no symmetry or definiteness assumption.
- **Its cost is no concern here.** GMRES's weakness is that it stores one vector per iteration. With an exact or near-exact LU preconditioner it converges in 1–2 iterations, so the storage is a handful of vectors.
- **It verifies the answer.** A factorisation with nudged pivots could silently return a wrong Newton step. GMRES measures the true residual and keeps iterating until it is below 10^{-10}, so a bad step is never passed to the Newton loop.

In short: the LU does almost all the work, and GMRES is the safety net that turns “almost” into a guaranteed-accurate solve.

B9. How GMRES works

GMRES builds an orthonormal basis $v_1, \ldots, v_k$ of the Krylov space by the **Arnoldi process**, a Gram–Schmidt orthogonalisation of Av_j against the previous vectors:

$$v_1 = \frac{b}{\beta},\ \beta = \|b\|; \qquad w = Av_j, \quad h_{ij} = w^{\mathrm{T}} v_i, \quad w \leftarrow w - h_{ij} v_i\ (i \le j), \quad h_{j+1,j} = \|w\|, \quad v_{j+1} = \frac{w}{h_{j+1,j}} \tag{B6}$$

The coefficients form an upper Hessenberg matrix $\bar{H}_k$ ($(k+1) \times k$) with $AV_k = V_{k+1}\bar{H}_k$. Writing $x = V_k y$, the residual norm becomes $\|b - AV_k y\| = \|\beta e_1 - \bar{H}_k y\|$, a small least-squares problem. It is solved incrementally with Givens rotations, which zero the subdiagonal one entry at a time; the last entry of the rotated right-hand side, $|g_{k+1}|$, is the residual norm, available at no cost each iteration. **Right preconditioning** applies GMRES to AM^{-1} (here

M^{-1} is the LU solve) and recovers $x = M^{-1}V_k y$, so the residual it minimises is still the true one.

Worked example: a 3 × 3 block

Take the 3 × 3 block of the coupled Jacobian belonging to node B2 (unknowns P_{B2}, h_{B2}, r_{B2}; rows: mass, heat, moisture), with the right-hand side $-F$ from the same iteration:

$$A = \begin{bmatrix} -1.0470 & -8.0418 & 18.874 \\ -1.6130\times10^{-4} & 0.99993 & 1.5850\times10^{-4} \\ -7.1215\times10^{-6} & -3.5410\times10^{-6} & 1.0000 \end{bmatrix}, \qquad b = \begin{bmatrix} -901.94 \\ 0.20963 \\ 9.0143\times10^{-3} \end{bmatrix}$$

Iteration 1. $\beta = \|b\| = 901.936$, $v_1 = b/\beta = \begin{bmatrix} -1.000000 \\ 2.32417\times10^{-4} \\ 9.99440\times10^{-6} \end{bmatrix}$. Then $w = Av_1$, $h_{11} = w^{\mathrm{T}}v_1 = -1.04528$, and after removing that component $h_{21} = \|w\| = 6.37240\times10^{-4}$. The least-squares problem $\min_y \|\beta e_1 - \bar{H}_1 y\|$ with $\bar{H}_1 = (h_{11}, h_{21})^{\mathrm{T}}$ gives $y_1 = \beta h_{11}/(h_{11}^2 + h_{21}^2) = -862.868$, $x_1 = y_1 v_1$, and a residual $\beta\, h_{21}/\sqrt{h_{11}^2 + h_{21}^2} = 0.54985$, i.e. 0.06 % of $\|b\|$.

Table 29. GMRES on the B2 block, no preconditioner.

k	h_{1k} … $h_{k+1,k}$ (Arnoldi column)	Residual $\|b - Ax_k\| / \|b\|$	x_k
1	−1.0453, 6.3724×10^{-4}	6.096×10^{-4}	862.87, −0.20055, -8.6238×10^{-3}
2	7.2184, 0.99826, 6.5692×10^{-8}	3.994×10^{-11}	859.08, 0.34822, 0.015133
3	19.204, -4.6290×10^{-3}, 1.0000, 5.8245×10^{-6}	3.080×10^{-19}	859.08, 0.34822, 0.015133

After three iterations the Krylov space is all of R^3 and the solution is exact: GMRES, like CG, terminates in at most n steps in exact arithmetic. The residual falls monotonically, as the minimisation property guarantees.

The full coupled system of case09

The whole first coupled iteration of case09 has 10 unknowns (P at the two free nodes, h and r at all four nodes). The table below shows GMRES on it with no preconditioner, with the diagonal (Jacobi) preconditioner, with the exact LU, and with the LU of a slightly wrong matrix (every diagonal entry 2 % too large, standing in for nudged pivots):

Table 30. Relative residual ‖b − Ax‖/‖b‖ after each GMRES iteration, case09 coupled system, iteration 1.

Iteration	No preconditioner	Jacobi	Exact LU	LU, 2 % perturbed
1	0.515	0.254	3.16×10^{-17}	3.69×10^{-3}
2	0.121	4.40×10^{-3}	—	9.09×10^{-6}
3	0.0368	1.04×10^{-3}	—	2.49×10^{-9}
4	6.37×10^{-5}	1.06×10^{-4}	—	2.06×10^{-11}
5	3.73×10^{-6}	4.53×10^{-6}	—	1.20×10^{-14}
6	4.04×10^{-16}	4.29×10^{-10}	—	—
7	—	1.26×10^{-16}	—	—

Without preconditioning GMRES needs 6 of the possible 10 iterations, because the rows have wildly different scales and the Krylov space must grow almost to the full dimension. The diagonal (Jacobi) preconditioner does not help here (7 iterations): it rescales the rows but cannot undo the strong one-way coupling between them. The exact LU preconditioner makes $AM^{-1} = I$, and GMRES finishes in one iteration at machine precision. The slightly wrong LU reaches the program's stopping test (10^{-10}) in 4 iterations. This is the regime the program runs in: on every coupled iteration of case09 GMRES took 1 iteration, with true relative residuals of 10^{-15}–10^{-17}.

For larger networks the same holds: GMRES restarts every 20 iterations and stops after 40 or at 10^{-10}, but with the envelope LU as preconditioner it is rarely asked for more than two.

Part C — The coupled airflow and thermal solver

C1. Why airflow and heat must be solved together

Part A treated the air temperature at every node as known. In a deep or hot mine it is not: the air warms as it descends (autocompression), takes heat and moisture from the rock, from machines and from people, and is cooled by coolers. The temperature, in turn, sets the density, and the density acts back on the airflow in four ways:

- resistance scales with density, $R = R_{\text{std}}\,\rho/\rho_{\text{ref}}$;
- the weight of every air column, $\rho g\,\Delta z$, and with it the natural ventilation pressure;
- the fan's pressure at its operating density;
- mass flow ρq, which is what is conserved at the nodes.

The heat, for its part, depends on the airflow: the mass flow carries it, and the air velocity sets the heat-transfer coefficient at the rock wall. So the two problems form one system of equations.

Solving one after the other (Picard iteration)

The straightforward approach alternates: solve the airflow with the current temperatures, then the temperatures with the new flows, and repeat until nothing changes. This is Picard (fixed-point) iteration. It converges when each round changes the answer by less than the round before, i.e. when the gain of one round, $|\partial q_{\text{new}}/\partial q_{\text{old}}|$, is below 1. Where a fan dominates, the temperatures change the flows only slightly, the gain is small, and Picard works.

Where natural ventilation drives the air, it fails. In a loop driven only by the density difference,

$$q = \sqrt{\frac{\text{NVP}}{R_{\text{loop}}}}, \qquad \text{NVP} = gH\,(\bar{\rho}_{\text{down}} - \bar{\rho}_{\text{up}}) \tag{C1}$$

so $\partial q/\partial\,\text{NVP} = 1/(2\sqrt{\text{NVP}\,R_{\text{loop}}})$, which grows without bound as the NVP, and the flow, become small. Meanwhile the NVP depends on the temperatures, and the temperatures on the flow: slow air in a hot shaft approaches the rock temperature. Each Picard round therefore swings the flow by the full amount. The program's first version moved the air after every airflow iteration and had exactly this fault: on large mines with big, low-resistance cross-cuts, a few pascals of natural ventilation swing hundreds of kg/s, and the airflow and the heat chased each other without end.

What the program does: settle the airflow, then the heat, and relax the air

The program keeps the two solvers apart, but changes the way they alternate. Each **round** (1) solves the airflow to convergence, with the densities of the current node air; (2) solves the heat on those flows; (3) moves the air at the nodes part of the way to the heat solve's

answer, by a step that adapts itself. Letting the airflow settle completely first, and relaxing the air between rounds, converges on mines where moving the air after every airflow iteration never settled. Where the relaxation stalls, an extra mixing term and Anderson acceleration take over (Section C5). The airflow solver used is a copy of the one of Part A that differs only in taking the air at each node into its densities; the airflow-only solve of Part A never sees a thermal result. This part describes the thermal equations, the iteration that joins them to the airflow, and follows it through case09 by hand.

The alternative: everything at once (Newton)

An all-at-once Newton formulation of the same problem, with the pressures *and* the air state at every node as unknowns and all cross-derivatives in one Jacobian, is also in the engine and is described in Section C8. It reaches the same answer on small networks, but it is not what the program runs: it converged on 4 of the 12 sample mines when last measured, against all 12 for the iteration of Section C5 (Section C7).

C2. The state of the air

At each node the air is described by three numbers: its pressure P (Pa), its specific enthalpy h (kJ per kg of dry air) and its moisture content r (kg of water vapour per kg of dry air). Enthalpy and moisture are the quantities that mixing and heat transfer conserve; temperatures follow from them:

$$h = 1.005\,t_{db} + r\,(2501 + 1.8\,t_{db}) \tag{C2}$$

so $t_{db} = (h - 2501r)/(1.005 + 1.8r)$; the wet bulb is found from r, P and t_{db} by inverting the psychrometer equation A5; the density follows from Eq. A6. If r exceeds saturation $r_s = 0.622\,e_s/(P - e_s)$ at the air's temperature, the excess condenses as mist (fog), keeping the enthalpy: the air is then saturated at a slightly higher temperature.

The moisture content follows from the vapour pressure as $r = 0.622\,p_w/(P - p_w)$, and the density of the moist air, in terms of the state, is

$$\rho = \frac{P}{287.05\,(t_{db} + 273.15)} \cdot \frac{1 + r}{1 + 1.6078\,r} \quad \text{(moist, default)}, \qquad \rho_{\text{dry}} = \frac{P}{287.05\,(t_{db} + 273.15)} \cdot \frac{}{1 +}$$

the same two bases as Eqs. A6 and A6b of Part A (the program's setting applies to both parts). **Hot, deep air.** The psychrometric routines shared with the airflow solvers clamp the dry bulb at 60 °C and the moisture at 50 g/kg. Deep, hot, wet air goes beyond that, and a fixed cap would hold it at 60 °C and drop the energy above it from the state. The heat solve therefore takes the state straight from (P, h, r) when either limit is passed: the dry bulb by bisection on Eq. C2, between −40 °C and as hot as the air needs, short of boiling at its pressure ($e_s < 0.95\,P$); the relative humidity as $p_w/e_s(t_{db})$; and the *thermodynamic* wet bulb t_{wb} as the temperature at which saturated air, less the water it has taken up, has the air's enthalpy:

$$h_{\text{sat}}(t_{wb}) - (r_{\text{sat}}(t_{wb}) - r)\,c_w\,t_{wb} = h, \qquad c_w = 4.187\ \text{kJ kg}^{-1}\text{K}^{-1}$$

C3. Heat along an airway

Along an airway the steady-flow energy equation per kilogram of air is

$$h_{\text{out}} = h_{\text{in}} + \frac{g\,(z_{\text{in}} - z_{\text{out}})}{1000} + \frac{\dot{Q}_{\text{added}}}{\dot{m}} \tag{C3}$$

with $\dot{Q}$ in kW and $\dot{m}$ in kg/s. The middle term is **autocompression**: air descending 1000 m gains $g \times 1000/1000 = 9.79$ kJ/kg at the default gravity of Section A5, about 9.7 °C of dry-bulb rise if nothing else happens (and loses it again rising). Friction has no separate term: the fan's work enters the air as pressure and friction turns it into heat further on, so the fan's shaft power is added as heat at the fan (a fixed-flow regulator that acts as a fan is treated the same way). Heat sources, coolers and the rock add to $\dot{Q}$; a source can also add water.

Heat and water from sources

Each airway's sources are applied at its midpoint, in kW, as $\Delta h = \dot{Q}/\dot{m}$:

- **Diesel engines:** $\dot{Q} = P_{\text{rated}}\,\text{LF}/\eta$ (load factor LF, default 1; efficiency η, default 0.33). Combustion also makes water, about 1.1 kg per kg of fuel, the fuel being $\dot{Q}$ divided by 45 MJ/kg: $\Delta r = 1.1\,\dot{Q}/(45\,000\,\dot{m})$. The latent heat of that water is part of the same $\dot{Q}$, so the enthalpy added is unchanged and only the moisture rises.
- **Electrical machines and distributed heat:** $\dot{Q} = \dot{Q}_{\text{machine}} + q'\,L$ (kW and kW per metre). A latent share λ (default 0, all dry heat) becomes water: $\Delta r = \lambda\,\dot{Q} \times 1000/(L_v(t_{db})\,\dot{m})$, again with the enthalpy added unchanged.
- **Blasting:** 3.7 MJ/kg of explosive, 20 % of it released as heat over the flushing time: $\dot{Q} = m_{\text{expl}} \times 3700 \times 0.2/\max(60,\ t_{\text{flush}})$ kW.

A source cannot heat the air without limit. In an airway that carries almost no air, as happens when the flow flips sign from round to round, $\dot{Q}/\dot{m}$ is thousands of kJ/kg and the rounds of Section C5 blow up. Electrical, diesel and blasting heat is therefore limited to 100 kJ/kg (about 100 °C of dry air): $\dot{Q} \le 100\,\dot{m}$. What does not fit is left out, and the report names the airways.

Rock heat: the Gibson/Starfield method

The rock around an airway starts at the **virgin rock temperature**, which rises with depth:

$$\text{VRT}(z) = \text{VRT}_{\text{datum}} + \frac{(z_{\text{collar}} - z - d_{\text{datum}}) \cdot \text{gradient}}{100} \tag{C4}$$

(gradient in °C per 100 m). Once ventilated, the rock near the wall cools and the heat flow falls with time. Modelling the airway as a cylinder of radius $r_0 = \text{Per}/2\pi$, the heat flux per square metre of wall after ventilation time t is

$$q = U\,(\text{VRT} - t_{\text{air}}), \qquad U = \frac{G(\text{Fo}, \text{Bi})\,k_r}{r_0}, \qquad \text{Fo} = \frac{\alpha t}{r_0^2},\ \ \text{Bi} = \frac{h_f r_0}{k_r},\ \ \alpha = \frac{k_r}{\rho_r c_r} \tag{C5}$$

where k_r, ρ_r, c_r are the rock's conductivity, density and specific heat, and G is Gibson's dimensionless heat flux. Instead of a curve fit, the program computes G from the exact solution for radial conduction outside a cylinder with a convective wall (Carslaw & Jaeger §13.5), whose Laplace transform

$$\bar{G}(p) = \frac{\mathrm{Bi}\,\sqrt{p}\,K_1(\sqrt{p})}{p\,(\sqrt{p}\,K_1(\sqrt{p}) + \mathrm{Bi}\,K_0(\sqrt{p}))} \tag{C6}$$

is inverted numerically by the Gaver–Stehfest algorithm (10 terms). The air-side coefficient comes from the airway's own friction factor by the Reynolds–Colburn analogy, $h_f = 0.71^{-2/3}\,k\,c_p\,|v|$ (at least 2 W/m²K for still air). **The calibrated model.** The program's default rock heat model is calibrated against a reference network: U is multiplied by a rock heat factor, 1.8 unless the site says otherwise (1 is the plain Gibson U), the walls are wet by the film model below, and there is no rock heat in a fan's own airway unless the airway asks for it. The examples of this guide that are checked by hand (case09, Section C6) use the plain model, factor 1 and the split wall, and say so.

Part of the wall is wet, by a wetness w on a rating scale: very dry 0, dry 0.1 (the default), damp 0.275, wet 0.55, very wet 0.85. The film model below is calibrated with the wetness taken at a tenth of its rating (dry 0.1 acts as 0.01), the plain split model with the rating as it stands; w below is the value the model in use takes. The dry part gives sensible heat, integrated exactly over each segment of length L_s since the air approaches the rock temperature:

$$\dot{Q}_s = \dot{m}\,c_p\,(\mathrm{VRT} - t_{db})\,(1 - e^{-(1-w)\,U\,\mathrm{Per}\,L_s/(\dot{m}c_p)}) \tag{C7}$$

Wet part, split model (the plain one). The wet part evaporates water: its heat $\dot{Q}_l = w\,U\,\mathrm{Per}\,L_s\,(\mathrm{VRT} - t_{wb})$ (the wet surface sits near the wet bulb) goes into latent heat, raising the moisture by $\Delta r = \dot{Q}_l/(\dot{m}\,(2501 - 2.383\,t_{wb}))$. The latent heat is capped so the air cannot pass saturation at the rock temperature: $\dot{Q}_l \le (h_{\mathrm{sat}}(\mathrm{VRT}) - h)\dot{m} - \dot{Q}_s$. The wet share gives the air no sensible cooling, so against a reference network this model leaves the air too hot and dry.

Wet part, film model (the default). The wet share is a film of water at its own surface temperature t_s. The rock conducts heat to the film, and the film gives it to the air by convection and by evaporation, with the same transfer effectiveness for both (the Lewis analogy), $\varepsilon = 1 - e^{-h_f A_w/(\dot{m}c_p)}$ over the wet area $A_w = w\,\mathrm{Per}\,L_s$:

$$U_r\,A_w\,(\mathrm{VRT} - t_s) = \dot{m}\,c_p\,\varepsilon\,(t_s - t_{db}) + L_v(t_s)\,\dot{m}\,\varepsilon\,(r_{\mathrm{sat}}(t_s) - r), \qquad U_r = \frac{U\,h_f}{h_f - U} \tag{C7a}$$

U_r is the rock's own conductance: Gibson's U is the rock and the air film in series, so the film is taken out. Conduction in falls and the film's losses rise with t_s, so Eq. C7a has one root, found by bisection (its upper limit is boiling at the air's pressure, so a runaway air temperature in a stub carrying almost no air cannot send the search far). The air then receives

$$\dot{Q}_{sw} = \dot{m}\, c_p\, \varepsilon\, (t_s - t_{db}), \qquad \dot{Q}_l = L_v(t_s)\, \dot{m}\, \varepsilon\, (r_{\text{sat}}(t_s) - r), \qquad \Delta r = \varepsilon\, (r_{\text{sat}}(t_s) - r)$$

$\dot{Q}_{sw}$ is negative when the film is cooler than the air: unlike the split model, a wet wall can *cool* dry air towards its wet bulb, as it does in a real heading. After each segment the enthalpy is $h \leftarrow h + (\dot{Q}_s + \dot{Q}_{sw} + \dot{Q}_l)/\dot{m}$, the moisture $r \leftarrow r + \Delta r$, and any excess over saturation condenses (Section C2). Airways are marched in segments of at most 50 m (at most 40 segments) so the heat follows the changing air temperature and rock temperature.

Coolers

A cooler is at the outlet of its airway and takes Δh kJ/kg out of the air. Four forms are modelled:

- **Spray (a direct-contact bulk air cooler; the default for a stated duty):** the air leaves saturated at the enthalpy that takes the duty out, $h_{\text{out}} = h - \dot{Q}_{\text{duty}}/\dot{m}$, gaining or losing water on the way: the leaving temperature t_a solves $h_{\text{sat}}(t_a) = h_{\text{out}}$ and $r_{\text{out}} = r_{\text{sat}}(t_a)$. Cooling at constant moisture instead left a benchmark surface cooler at the wrong state (22.5 °C dry bulb and 47 % RH, where the reference has 15.2 °C dry and 15.1 °C wet bulb).
- **Coil:** the air moves towards the coil-surface state, 0.9 of the way (the contact factor). With the apparent dew point t_a from $h_{\text{sat}}(t_a) = h - \Delta h/0.9$, the outlet moisture is $r_{\text{out}} = r + 0.9\, (r_{\text{sat}}(t_a) - r)$ if that is below r (it condenses), otherwise r (a dry coil only cools), and $h_{\text{out}} = h - \Delta h$.
- **Bulk air cooler with water:** given the water's inlet temperature $t_{w,in}$, flow $\dot{m}_w$ and an effectiveness ϵ_c (default 0.8), $\Delta h = \epsilon_c\, (h - h_{\text{sat}}(t_{w,in}))$, the air leaving as for the spray; the water warms by $\dot{Q}/(\dot{m}_w\, 4.186)$ and the duty is held back where the water would leave warmer than the air's own dry bulb arriving (it would then heat the air).
- **Target-state cooler:** the air is taken to a stated dry and wet bulb when that is drier than the air arriving, otherwise to the stated dry bulb at the same moisture (a coil cannot add water), limited by a capacity in kW; it never heats.

Validation: the Bluebook's tunnel example

The Bluebook (du Plessis 2014, ch. 22, Example 2) works the heat flow into a 3 m diameter circular tunnel in rock with $k_r = 6$ W/m°C and $\alpha = 2.5 \times 10^{-6}$ m²/s, ventilated for 4 years, with a virgin rock temperature of 44 °C. Then $r_0 = 1.5$ m and $\text{Fo} = 2.5 \times 10^{-6} \times 4 \times 3.156 \times 10^7/1.5^2 = 140.3$. The heat flow per metre of tunnel is $q' = 2\pi k_r\, G\, (\text{VRT} - t_{\text{eq}})$. The book reads its values from Starfield's and Goch & Patterson's tables; ASM Lab computes G by Eq. C6:

Table 31. Rock heat: ASM Lab against the Bluebook's worked example. The same comparison runs in the program's test suite.

Case	h (W/m²K)	t_eq (°C)	Bi	G (ASM Lab)	Heat flow, ASM Lab (W/m)	Heat flow, Bluebook (W/m)
Wet surface	59.3	26.9	14.82	0.3211	207.0	207
Dry surface	14	32	3.50	0.3012	136.3	136
Dry, insulated	0.79	32	0.20	0.1250	56.5	57
Bi → ∞ (no film)	—	—	∞	0.3277	—	T = 0.327 (Goch & Patterson)

All three cases agree to the precision the book quotes. The psychrometric constants of Sections A4 and C2 are those of the Blackbook (PEE 410, ch. 5): e_s with 0.6105 kPa, $H'_w = 1.8\,t + 2501$ kJ/kg, $\gamma = 0.000644$, and the latent heat $2501 - 2.383\,t$ of the Bluebook (ch. 19).

C4. Mixing at nodes

At a junction the incoming streams mix. Their mass flows $\dot{m}_e$ add, and so do their enthalpy and moisture flows:

$$h_n = \frac{\sum_{e\to n} |\dot{m}_e|\, h_{\text{out},e} + M_{\text{ext}}\, h_{\text{ext}}}{\sum_{e\to n} |\dot{m}_e| + M_{\text{ext}}}, \qquad r_n = \text{same with } r \tag{C8}$$

where the sum is over the airways delivering air *into* n (whichever way they are drawn), and M_{ext} is outside air a boundary node draws in (the air leaving it minus the air arriving), at the node's stated dry and wet bulb or the site's surface conditions.

Sweeps along the flow. The heat solve visits the nodes in the order of the flow, sources first (the intakes and exhausts that air enters from, and any junction that air only leaves), marches each airway from the air at its upstream node by Section C3, and mixes at the downstream node by Eq. C8. Without recirculation one sweep is exact; where air returns round a loop the sweeps are repeated until no node's air changes by more than 10^{-9} kJ/kg, or by less energy than 10^{-6} of the largest energy flow ($10^{-6}\,\dot{m}_{\text{peak}}$ kW): a slow loop of nearly still air creeps by tiny amounts that move no energy. After the first sweep only the nodes whose incoming air changed are revisited, and an airway is marched again only when the air entering it changed.

Still air. A node that no air reaches has no mix to take (0/0), and where the flow is only the airflow solve's tolerance its sign is noise. Such a node, a dead end, a stub or an airway with no flow, takes the *still-air* state. It is also what an airway gives when almost no air passes, so that no flow and next to no flow agree: where the rock heats the node's airways, air at the rock temperature (saturated if the wall is wet); where it does not, the mean air of the neighbours that have air. Air that only circles round a loop that nothing enters is still air too: with no fresh air it settles to the rock, and sweeping it lap after lap creeps for thousands of passes.

C5. The coupled iteration

Write x for the air at every node: the enthalpy h_n and the moisture r_n, the moisture carried as 2.5 times its value in g/kg so that 1 g/kg weighs as much as the 2.5 kJ/kg of its latent heat. One **round** is a map $x \mapsto g(x)$: solve the airflow with the densities of x, solve the heat on the flows that gives, and read the node air off the heat solve. The coupled solution is the fixed point $x = g(x)$; the round's *change* is $d = g(x) - x$.

1. **Round 0.** Solve the airflow with the air the nodes state (a node that states none takes the site's surface air, Eq. A6a) and the heat on those flows. That gives the first x. If this airflow does not converge nothing starts.
2. **Airflow.** The Newton solve of Part A, on the copy of the airflow solver, with the node air of x turned into densities and started from the pressures of the last round, so later rounds take few iterations. This copy differs from Part A's solver in one more way: its friction has a *laminar region*, $\Delta P = R\,q\sqrt{q^2 + q_L^2}$, where $q_L = 2000\,\mu A/(\rho\,D_h)$ is the flow at Reynolds number 2000 in that airway ($\mu = 1.81 \times 10^{-5}$ Pa·s). At real flows $q \gg q_L$ it is the square law to within $(q_L/q)^2/2$ (0.007 % at 10 m³/s in a 4 m drift); through zero it is the straight line $R\,q_L\,q$, so the flow responds to a pressure difference with a finite slope where the square law's is infinite (Section A10). It is what lets the coupled rounds pass through an airway that hovers about zero flow.
3. **Heat.** The sweeps of Section C4 on those flows, started from the last round's outlet air. Their node air is $g(x)$.
4. **Test** (Eq. C12). If the change is within tolerance the rounds stop.
5. **Step.** Move x towards $g(x)$ by Aitken's rule, or by Anderson's once the rounds have stalled.

Convergence is judged on the energy and the water the air carries. A node that almost no air passes (a dead end, a stagnant heading) can creep by tenths of a kJ/kg without moving a watt or the airflow, so each node's change is weighed by the air passing through it, M_n (half the sum of the mass flows of its airways), with d_h in kJ/kg and d_r, g_r in g/kg:

$$\begin{aligned} \max_n M_n\,|d_{h,n}| &\le \max(10^{-3},\ 10^{-6}\max_n M_n\,|g_{h,n}|)\ \text{kW} \\ \max_n M_n\,|d_{r,n}| &\le \max(10^{-3},\ 10^{-6}\max_n M_n\,|g_{r,n}|)\ \text{g/s} \end{aligned} \tag{C12}$$

The round's reported *change* is the larger of the two ratios, so it is at most 1 when the rounds have settled.

Aitken's step. The first rounds move the air a share ω of the way to the heat solve's, $x \leftarrow x + \omega\,d$. It starts at $\omega = 0.5$ and is adapted from the last two changes: large while successive changes point the same way, small where they overshoot each other.

$$\omega_k = \text{clip}\!\left(-\,\omega_{k-1}\,\frac{d_{k-1}\cdot(d_k - d_{k-1})}{\|d_k - d_{k-1}\|^2},\ 0.02,\ 1\right) \tag{C13}$$

When the rounds stall. If ω sits at its floor for five rounds in a row, or round 40 is reached, some airways are hovering about zero flow, held there by natural ventilation, and flip direction from round to round. The air at their ends then jumps between the two ends' air and there is no smooth answer to settle to. Two things change from there. First, **exchange**: a nearly still airway away from the intakes and exhausts carries, besides its net flow m, an exchange flow x_e in each direction,

$$x_e = \frac{\epsilon^2}{\epsilon + |m|}, \qquad \epsilon = 3 \times 10^{-3}\, \dot{m}_{\text{peak}} \quad (\text{left out beyond } |m| = 20\,\epsilon) \tag{C14}$$

The two flows are equal, so the mass is unchanged and the energy is only mixed: the air at the far end passes smoothly through zero net flow, instead of jumping from one end's air to the other's, as real still air does. Where air moves, $|m| \gg \epsilon$ and the exchange is next to nothing. Mines that settle before the stall never see it. Second, **Anderson acceleration**: the step is taken from the last six rounds. With Δx_j and Δd_j the differences of x and d over successive rounds, a multisecant estimate of how the airflow and the heat answer each other (the Jacobian of the whole coupled system, found without forming it) gives the weights and the step:

$$\gamma = \arg\min_{\gamma} \left\| W(d - \sum_j \gamma_j\, \Delta d_j) \right\|, \quad W = \operatorname{diag}(\sqrt{M_n}), \qquad x \leftarrow x + \beta\, d - \sum_j \gamma_j\, (\Delta x_j + \beta\, \Delta d_j), \quad \beta = 0.6 \tag{C15}$$

The weight W makes the fit follow the energy the air carries, as Eq. C12 does. The normal equations are lightly regularised. A node's move is limited to $4\,|d| + 0.1$ kJ/kg, no further than a few plain rounds would take it, and a node through which almost no air passes (weight below 1 % of the largest) takes a plain step $\beta\, d$, because it barely shows in the fit and must not be thrown off. When a round is ten times worse than the best since the last reset, the history no longer describes where the iteration is: it is forgotten and a plain step of 0.30 is taken. Anderson's step finds the balance that a loop of warm, buoyant air settles to, which plain or damped rounds can circle round without reaching.

When the airflow will not follow. If the airflow solve does not converge with the air a step produced, the iteration goes back to the air before that step, takes a short plain step of 0.25 of the change from there, and forgets the Anderson history. After three such failures in a row it stops and says so.

If the rounds do not settle (at most 150), the program does not show the last round, which can be the worst one: on one benchmark mine it ended on a step that took the change from about 100 to 10^5. It shows the round with the smallest change, marked as not converged, with how far from settled it is; and, if that round is not sound, the airflow solved with the declared air followed by the heat on those flows (one pass), without the natural ventilation the heat would drive.

The report. After the last round the program lists the energy balance in kW: rock heat (sensible and latent), autocompression (net, down less up), diesel, electrical, blasting, fans

and coolers, against the energy the air carries out less what it carries in. It flags an error if the balance does not close within its tolerance or the sweeps did not settle. It also states the rock heat model in use (wall model, wetness, rock heat factor), so a result can be traced to the model that made it.

C6. Worked example — natural ventilation with real temperatures (case09)

The network of Section A16 (two 500 m shafts and a 400 m level), now with the heat computed rather than imposed. Only the intake S1 keeps its stated air (5 °C, saturated); the temperatures at B1, B2 and S2 are unknowns. The site is the program's "SA Bushveld (Platinum)" geothermal profile, with the plain rock heat model (rock heat factor 1, split wet wall) so that every number can be checked by hand:

Table 32. case09: thermal site data.

Parameter	Value
VRT at datum	38 °C at 1500 m below the collar
Geothermal gradient	1.2 °C per 100 m
Rock k, ρ, c	2.8 W/m·K, 2750 kg/m³, 820 J/kg·K
Age of workings (default)	5 years
Wall model	split (plain); wetness 0.1, i.e. 10 % of the wall wet
Rock heat factor	1 (the plain Gibson U)
Surface air	22 °C dry bulb, 15 °C wet bulb, 101.325 kPa

Rock temperatures. By Eq. C4, at the mid-depth of the shafts (250 m) $\mathrm{VRT} = 38 + (250 - 1500) \times 1.2/100 = 23.0$ °C, and on the level (500 m) $38 + (500 - 1500) \times 1.2/100 = 26.0$ °C.

The rounds

Only the intake S1 states its air (5 °C, saturated). B1, B2 and S2 state nothing, so in round 0 they take the site's surface air (22 °C dry bulb, 15 °C wet bulb, 101.325 kPa, carried to their elevation by Eq. A6a): a downcast of 5 °C air and an upcast of surface-warm air, which drives 114.3 m³/s round the loop, far more than the rock will let the air sustain. Each round then recomputes the air from the heat the rock gives on the last flows:

Table 33. case09: every round, with the plain rock heat model. Flows are the round's airflow solve; temperatures are the heat solve's air on those flows (dry bulb). “Change” is Eq. C12, so 1 is settled; “Airflow it.” is the number of Newton iterations of that round's airflow solve.

Round	Downcast (m³/s)	Level (m³/s)	Upcast (m³/s)	B1 (°C)	B2 (°C)	S2 (°C)	Change (× tol.)	Aitken ω	Airflow it.
0 (start)	114.25	113.55	116.92	10.226	10.535	6.108	—	—	—
1	38.52	37.76	38.62	10.950	11.786	7.979	55770	0.50	6
2	52.28	51.30	52.53	10.668	11.304	7.270	6499	0.57	5
3	53.87	52.87	54.15	10.645	11.263	7.209	949	0.67	4
4	54.18	53.17	54.45	10.640	11.256	7.198	33.2	0.67	3
5	54.18	53.18	54.46	10.640	11.255	7.197	18.4	0.85	2
6	54.18	53.18	54.46	10.640	11.255	7.197	3.04	1.00	2
7	54.18	53.18	54.46	10.640	11.255	7.197	0.0871	0.97	2

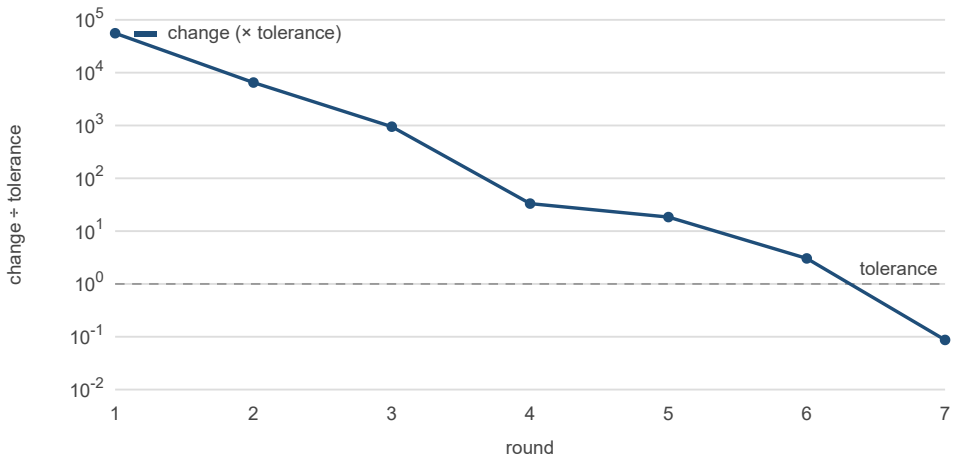


Figure 9. case09: the change after each round, against its tolerance of 1.

Round 0 and round 1. The heat solve on the 114.3 m³/s flow cools the upcast to 6.11 °C at S2: fast air has little time to take heat from the rock. The airflow with that cooler air has a smaller natural ventilation pressure and falls to 38.5 m³/s, below the answer. The slow air then takes more heat per kilogram, and the heat solve gives 7.98 °C at S2, above it. The two solvers, taken in turn, overshoot each other, which is the swing of Section C1; a plain Picard step would carry on swinging.

Rounds 2 to 7. Aitken's rule starts ω at 0.5 and raises it as successive changes line up (0.50, 0.57, 0.67, 0.67, 0.85, 1.00, 0.97), and the flow settles: 38.524 → 52.277 → 53.875 → 54.177 → 54.184 → 54.185 → 54.185 m³/s. The change falls 55770 → 6499 → 949 → 33.2 → 18.4 → 3.04 → 0.0871 times the tolerance. Once ω reaches 1 the step is the heat solve's own answer and the iteration converges like Newton's method, a few digits per round. No stall occurs, so neither the exchange nor Anderson's step is used.

The state it settles to is the one the all-at-once Newton formulation of Section C8 reaches in eight iterations: the node temperatures, pressures, moistures and flows below agree with it to every digit shown.

The converged state

Table 34. case09 coupled: converged node states.

Node	z (m)	P (Pa)	t_{db} (°C)	t_{wb} (°C)	r (g/kg)	h (kJ/kg)	ρ (kg/m³)
S1	0	101325.00	5.000	5.000	5.3986	18.5756	1.26497
B1	-500	107637.98	10.640	8.306	5.4389	24.4002	1.31704
B2	-500	107618.94	11.255	8.639	5.4716	25.1070	1.31394
S2	0	101325.00	7.197	6.255	5.5106	21.0866	1.25497

Table 35. case09 coupled: airways, compared with the airflow-only solve of Section A16.

Airway	q coupled (m³/s)	ṁ (kg/s)	q airflow-only (m³/s)	Autocompression (kW)	Rock sensible (kW)	Rock latent (kW)
down	54.18	69.953	242.58	342.42	58.03	7.00
level	53.18	69.953	248.99	0.00	43.77	5.67
up	54.46	69.953	271.61	−342.42	54.41	6.78

Checking the downcast shaft by hand

Air velocity and film coefficient. $v = q/A = 54.185/36 = 1.5051$ m/s; with the shaft's $k = 0.008$: $h_f = 0.71^{-2/3} \times 0.008 \times 1005 \times 1.5051 = 15.205$ W/m²K.

Gibson parameters. $r_0 = 24/2\pi = 3.8197$ m; $\alpha = 2.8/(2750 \times 820) = 1.2417 \times 10^{-6}$ m²/s; $\mathrm{Fo} = 1.2417 \times 10^{-6} \times 5 \times 3.156 \times 10^{7}/3.8197^2 = 13.428$; $\mathrm{Bi} = 15.205 \times 3.8197/2.8 = 20.743$. The Stehfest inversion of Eq. C6 gives $G = 0.4906$, so $U = 0.4906 \times 2.8/3.8197 = 0.3596$ W/m²K and, over the whole wall, $U\,\mathrm{Per}\,L = 0.3596 \times 24 \times 500 = 4316$ W/K.

Autocompression. $\dot{m}\,g\,\Delta z/1000 = 69.953 \times 9.79 \times 500/1000 = 342.42$ kW (program: 342.42 kW).

Rock heat, one-step estimate. With the mean air temperature in the shaft, $\bar{t}_{db} = (5.00 + 10.64)/2 = 7.82$ °C and $\bar{t}_{wb} = 6.65$ °C, against the mid-depth VRT of 23 °C:

$$\dot{Q}_s \approx 0.9 \times 4316 \times (23 - 7.82) = 58.96\ \mathrm{kW}, \qquad \dot{Q}_l \approx 0.1 \times 4316 \times (23 - 6.65) = 7.05\ \mathrm{kW}$$

against the program's 58.03 and 7.00 kW. The program's figures differ by a few per cent because it marches 10 segments of 50 m, each with its own rock temperature (20.3 °C at the top, 25.7 °C at the bottom) and the air temperature reached there.

Energy equation C3. $h_{B1} = 18.5756 + \frac{342.42+58.03+7.00}{69.953} = 18.5756 + 5.8246 = 24.4002$ kJ/kg; the solved enthalpy at B1 is 24.4002. Since only the downcast delivers air to B1, Eq. C8 makes h_{B1} equal to the air the shaft delivers.

Moisture. $\Delta r = \dot{Q}_l/(\dot{m}\,(2501 - 2.383\,\bar{t}_{wb})) = 6.997/(69.953 \times 2485.1) = 4.025 \times 10^{-5}$ kg/kg $= 0.0402$ g/kg, so $r_{B1} = 5.3986 + 0.0402 = 5.4389$ g/kg (solved: 5.4389).

Why the flow fell from 243 to 54 m³/s

The natural ventilation pressure. Mean densities: downcast $(1.26497 + 1.31704)/2 = 1.29101$, upcast $(1.31394 + 1.25497)/2 = 1.28445$ kg/m³. So $\mathrm{NVP} = 9.79 \times 500 \times (1.29101 - 1.28445) = 32.08$ Pa.

The friction it overcomes. $R_{\mathrm{down}} = R_{\mathrm{up}} = 0.008 \times 500 \times 24/6^6 = 2.058 \times 10^{-3}$ and $R_{\mathrm{level}} = 0.012 \times 400 \times 20/5^6 = 6.144 \times 10^{-3}$ N·s²/m⁸ at 1.2 kg/m³, scaled by density: $\sum Rq^2 = 2.2137 \times 10^{-3} \times 54.18^2 + 6.7353 \times 10^{-3} \times 53.18^2 + 2.2024 \times 10^{-3} \times 54.46^2 = 32.08$ Pa. The loop balances.

The airflow-only solve took the upcast at the 35 °C written on its nodes, an NVP of about 680 Pa and 243 m³/s. With the heat computed, the rock (20–26 °C) warms the air only to 11.3 °C at the bottom of the upcast, which cools again to 7.2 °C by expansion as it rises. The density difference, and the NVP, are about 20 times smaller, and the flow $\propto \sqrt{\mathrm{NVP}}$ about 4.5 times smaller. Imposed temperatures can make natural ventilation look far stronger than the rock and the air can sustain; the coupled solve gives the self-consistent answer.

The energy balance

All energy entering the network must leave it. Air enters at S1 and leaves at S2 at the same elevation, so the potential-energy terms cancel and $\dot{m}\,(h_{S2} - h_{S1}) = 69.953 \times (21.0866 - 18.5756) = 175.65$ kW. The rock adds 175.65 kW (sensible + latent, all three airways), and autocompression down and expansion up cancel (342.42 and −342.42 kW). The balance closes to 8.92\times10^{-5} kW; the program reports it to 10⁻¹² kW, because it sums the energy flows exactly rather than from rounded node values.

The same network with the program's default rock heat model

The hand checks above use the plain model (rock heat factor 1, split wall). With the program's defaults, rock heat factor 1.8 and film walls with the dry rating 0.1 acting as 0.01, the rock gives the air 282.9 kW sensible and 18.3 kW latent instead of 156.2 and 19.4 kW. The upcast is warmer, the natural ventilation pressure larger, and the loop settles in 7 rounds at 65.7 m³/s instead of 54.2:

Table 36. case09 with the plain and the default rock heat models. The default model is calibrated against a reference network, not against case09; the table shows what the choice of model is worth on one network.

Node	t_{db}, plain (°C)	t_{db}, default (°C)	r, plain (g/kg)	r, default (g/kg)	P, default (Pa)
S1	5.000	5.000	5.399	5.399	101325.0
B1	10.640	11.116	5.439	5.416	107629.3
B2	11.255	11.987	5.472	5.460	107601.4
S2	7.197	8.290	5.511	5.486	101325.0

C7. Convergence on the sample mines

The iteration of Section C5 was run on the program's 12 sample mines with its default settings and the SA Bushveld (Platinum) site profile (each sample's own site data where it has any): the same mines and site on which the all-at-once Newton formulation of Section C8 was measured. It settles on **11 of the 12**; the Newton formulation settled on 4.

Table 37. The airflow + heat iteration on the sample mines (SA Bushveld site profile, default rock heat model). The last column is how far the flows moved when the heat was allowed to act on them, against the largest flow of the airflow-only solve.

Sample	Size	Airflow only	Airflow + heat	Rounds	Dry bulb range (°C)	Largest change in an airway's mass flow (kg/s)
copper_porphyry	66 nodes, 113 airways	converged (0.09 s)	converged (0.26 s)	5	23.3 – 28.5	9.15 of 182.7
ug2_reef	51 nodes, 59 airways	converged (0.01 s)	converged (0.09 s)	4	21.4 – 26.7	1.96 of 74.8
merensky_reef	46 nodes, 63 airways	converged (0.13 s)	converged (0.10 s)	5	22.4 – 27.3	3.77 of 94.8
coal_seams	660 nodes, 1216 airways	converged (0.81 s)	converged (89.04 s), exchange + Anderson	34	20.3 – 26.5	14.10 of 181.5
gold_vein	142 nodes, 159 airways	converged (0.08 s)	converged (11.34 s), exchange + Anderson	45	21.3 – 26.5	2.20 of 79.1
nickel_massive	24 nodes, 27 airways	converged (0.03 s)	converged (0.09 s)	5	19.7 – 26.6	0.55 of 99.6
Untitled_Mine	7 nodes, 7 airways	converged (0.00 s)	converged (0.01 s)	3	22.0 – 26.5	10.73 of 506.0
sample_test1	29 nodes, 35 airways	converged (0.02 s)	converged (0.07 s)	5	20.1 – 26.0	1.59 of 54.5
sample_test2	11 nodes, 13 airways	converged (0.00 s)	did not settle (0.92 s), exchange + Anderson	150 (limit)	23.2 – 26.7	8.55 of 77.4
sample_test3	14 nodes, 16 airways	converged (0.01 s)	converged (0.05 s)	5	26.0 – 64.8	16.91 of 147.1
sample_test4	17 nodes, 21 airways	converged (0.01 s)	converged (0.18 s)	19	19.1 – 40.4	14.52 of 146.4
sample_test5	26 nodes, 28 airways	converged (0.00 s)	converged (0.02 s)	5	20.3 – 31.3	1.91 of 49.7

3 mines (coal_seams, gold_vein, sample_test2) stalled under plain Aitken rounds and went on to the exchange and Anderson's step of Section C5; the others never needed them. The largest, coal_seams with 1216 airways, took 89 s in this run, against 0.8 s for its airflow-only solve: every round is a full airflow solve and a full heat solve, so a large mine takes a minute or more, not seconds. On sample_test3 the heat moves an airway's flow by 11 % of the largest flow, which is why the coupled answer matters.

sample_test2 did not settle. After 150 rounds the energy the air carries still changed by 341 times its tolerance between rounds. The program shows the round with the smallest change, marked as not converged (Section C5), and says how far from settled it is. The

same network settles on the Witwatersrand (Gold) profile, in 53 rounds, so whether a loop settles depends on the thermal conditions as well as on the network.

C8. The all-at-once Newton formulation (in the engine, not the default)

Earlier versions of this guide presented this formulation as the coupled solver. It is still in the engine, and the machinery it needs is the subject of Parts B7–B9, but the program does not run it: Section C5 is what runs. It is kept here because it reaches the same answer on small networks (case09 above agrees digit for digit) and because it shows the coupling as a single system of equations, which is how it would be solved by a method with every derivative available.

Mixing with a still-air tie

Still air. A node that no air reaches has no mixing equation: 0/0. The all-at-once formulation adds a small tie to a “still air” state, the rock-temperature air where the rock heats the node's airways (saturated if the rock is wet), otherwise the mean of the neighbours:

$$h_n = \frac{\sum |\dot m_e|\, h_{\mathrm{out},e} + M_{\mathrm{ext}} h_{\mathrm{ext}} + \tau\, h_{\mathrm{still}}}{\sum |\dot m_e| + M_{\mathrm{ext}} + \tau}, \qquad \tau = \frac{\varepsilon^2}{M + \varepsilon}, \quad \varepsilon = 10^{-7} \max_e |\dot m_e| \tag{C9}$$

with M the air arriving. Where air flows ($M \gg \varepsilon$) the tie is $\approx \varepsilon^2/M$, utterly negligible; where none arrives it is ε and decides the state. The tie also makes the equations continuous as a flow passes through zero, which Newton's method needs.

The coupled equations

The unknowns are the pressure at every free node and the enthalpy and moisture at *every* node (boundary nodes included: their pressure is fixed, but the air there is not, because an exhaust receives whatever arrives). At each node:

$$\begin{aligned} F_n^{\mathrm{mass}} &= \textstyle\sum_{\mathrm{in}} \rho_e q_e - \sum_{\mathrm{out}} \rho_e q_e = 0 && \text{(free nodes; kg/s)} \\ F_n^{\mathrm{heat}} &= h_n - h_{\mathrm{mix},n} = 0 && \text{(kJ/kg)} \\ F_n^{\mathrm{moist}} &= 1000\,(r_n - r_{\mathrm{fog}}(h_{\mathrm{mix},n}, r_{\mathrm{mix},n})) = 0 && \text{(g/kg)} \end{aligned} \tag{C10}$$

where $h_{\mathrm{mix}}, r_{\mathrm{mix}}$ are the right-hand sides of Eq. C9, computed from the air *delivered* by each inflowing airway, i.e. its upstream node's air marched along it by Section C3. The densities in the mass equation come from each node's own P, h, r. The airflow equations are exactly those of Part A, with the temperatures now unknowns rather than inputs. Moisture is carried in g/kg so the three rows have comparable sizes.

Tolerances. Mass as in Part A (10^{-6} of the largest mass flow, at least 10^{-4} kg/s), and 10^{-6} kJ/kg for enthalpy and 10^{-6} g/kg for moisture: tight enough that the energy balance of a whole mine closes to about a watt (800 kg/s × 10^{-6} kJ/kg = 0.8 W).

The coupled Jacobian

The Jacobian has a 3 × 3 block for each pair of neighbouring nodes:

$$J_{nk} = \begin{bmatrix} \partial F_n^{\text{mass}}/\partial P_k & \partial F_n^{\text{mass}}/\partial h_k & \partial F_n^{\text{mass}}/\partial r_k \\ \partial F_n^{\text{heat}}/\partial P_k & \partial F_n^{\text{heat}}/\partial h_k & \partial F_n^{\text{heat}}/\partial r_k \\ \partial F_n^{\text{moist}}/\partial P_k & \partial F_n^{\text{moist}}/\partial h_k & \partial F_n^{\text{moist}}/\partial r_k \end{bmatrix} \tag{C11}$$

- **Mass rows, pressure columns:** analytically, exactly as in Part A (Eq. A15).
- **Mass rows, h and r columns:** how a node's air changes the density of its airways, hence their resistance, column weight and mass flow. This is the natural-ventilation coupling.
- **Heat and moisture rows:** $\partial F/\partial h_n = 1$ on the diagonal, $-\partial h_{\text{mix}}/\partial h_k$ for each upstream neighbour k (the share of k's air in the mix, modified by the march), and pressure columns through the mass flows that weight the mix.

Everything except the analytic block is computed by **local finite differences**: nudge one unknown at one node ($\delta h = 10^{-4}\max(1,|h|)$, $\delta r = 10^{-7}$, $\delta P = \max(10^{-4}, 10^{-8}P)$), recompute that node's density, its airways (flow and delivered air) and the equations at the node and its neighbours, and divide the changes by the nudge. Only the neighbourhood is recomputed, so building the Jacobian costs a few airway marches per node, not a full solve per unknown.

The matrix is not symmetric (Part B7): heat flows downstream only. The unknowns are numbered node by node in RCM order with each node's P, h, r together, so each 3 × 3 block sits on or near the diagonal and the envelope LU of Part B8 stays small, with GMRES to finish (Part B9).

The coupled Newton loop

1. **Start** from the airflow-only solution and a thermal solve on it (Picard's first round), so the first Newton step starts close.
2. **Evaluate** every node's temperatures and density from (P, h, r), every airway's flow (Part A), mass flow and delivered air (Section C3), and all residuals (Eq. C10).
3. **Build** the Jacobian (Section C6) and **solve** $J\,\delta x = -F$ by envelope LU + GMRES.
4. **Line search** on the scaled merit function $\Phi = \sum_i (F_i/\text{tol}_i)^2$: accept α if $\Phi(x + \alpha\,\delta x) \le (1 - 10^{-4}\alpha)\,\Phi(x)$, else halve, up to 12 times (if none passes, the best trial is taken).
5. **Stop** when every residual is within its tolerance ($\max_i |F_i|/\text{tol}_i \le 1$). As in Part A, the residual stagnating within 10× of tolerance for 6 iterations is reported as the precision limit, and no progress for 15 iterations as a stall.

Status when it was measured

This formulation converges on case09, as shown above, but of the 12 sample mines it converged on only 4 when it was measured (these figures are from that measurement and have not been repeated). On the others the line search stalls, and on coal_seams the iterates diverge. It is the reason the program couples airflow and heat by the iteration of Section C5 instead.

Table 38. The all-at-once Newton formulation on the sample mines (SA Bushveld site profile), as measured when it was written.

Sample	Size	Airflow only	Coupled
copper_porphyry	66 nodes, 113 airways	converged (0.1 s)	did not converge, 15 iter. (0.2 s)
ug2_reef	51 nodes, 59 airways	converged (0.0 s)	converged, 4 iter. (0.0 s)
merensky_reef	46 nodes, 63 airways	converged (0.0 s)	converged, 9 iter. (0.0 s)
coal_seams	660 nodes, 1216 airways	converged (0.2 s)	did not converge, 17 iter. (2.8 s)
gold_vein	142 nodes, 159 airways	converged (0.0 s)	did not converge, 15 iter. (0.2 s)
nickel_massive	24 nodes, 27 airways	converged (0.0 s)	did not converge, 15 iter. (0.0 s)
Untitled_Mine	7 nodes, 7 airways	converged (0.0 s)	converged, 2 iter. (0.0 s)
sample_test1	29 nodes, 35 airways	converged (0.0 s)	converged, 25 iter. (0.1 s)
sample_test2	11 nodes, 13 airways	converged (0.0 s)	did not converge, 20 iter. (0.0 s)
sample_test3	14 nodes, 16 airways	converged (0.0 s)	did not converge, 24 iter. (0.1 s)
sample_test4	17 nodes, 21 airways	converged (0.0 s)	did not converge, 33 iter. (0.4 s)
sample_test5	26 nodes, 28 airways	converged (0.0 s)	did not converge, 22 iter. (0.1 s)

The failures share one cause, and the balanced bridge (lit2) reproduces it with four nodes: an airway at (nearly) zero flow. Part A's airflow solver passes through that point thanks to its line search on the largest mass residual (Section A10). In the coupled system the zero-flow airway also switches which node is upstream, and therefore which node's air the heat equations draw on. The analytic pressure slope there (capped at 10^6) and the finite-difference heat derivatives describe the same point very differently, so the Newton direction is poor and the merit function, dominated by the tight heat tolerances, accepts only tiny steps. The iteration of Section C5 avoids this by solving the airflow to convergence in every round, and the exchange of Eq. C14 smooths the switch of upstream node as a flow passes through zero.

Part D — The fire simulation

D1. What it computes

The fire simulation follows a mine fire *in time* over the ventilation network: where the heat and the smoke go, how hot the air gets, how the fire's heat changes the airflow, and what happens when a fan stops or a door is closed. It is a one-dimensional model after the classic network fire programs of the US Bureau of Mines: **plug flow** along each airway, **perfect mixing** at the junctions, heat and fume treated like a contaminant carried by the air, heat exchange with the rock, and the airflow solved again with the fire's air temperatures, so that throttling and natural draft follow from density.

It is a different calculation from Part C. Part C finds a steady state in which airflow and heat agree. The fire model marches through time, and its airflow is a quasi-steady solve repeated every few seconds on the changing densities. It runs on its own copy of the airflow solver, so that work on the fire never touches the tunnel airflow and thermal solves.

What it does *not* do is as important as what it does: it has no roof layer, no smoke running upstream along the roof (the program flags the conditions where this is likely, Section D8), no flame spread and no ducts. Section D10 gives measured evidence for how far its numbers can be trusted, and where they cannot.

D2. Plug flow along an airway

Each airway holds a list of *parcels* of air in order along its length. A parcel has a volume v, a temperature t, a fume fraction ϕ (percent of the air that is fume) and a concentration of each tracked gas or dust. Every time step Δt (5 s by default) the parcels move along: the volume that leaves the airway at its downstream end is the flow's own,

$$v_{\text{out}} = |q|\,\Delta t + \max(0,\ v_{\text{held}} - v_{\text{cap}}), \qquad v_{\text{cap}} = A\,L\,\frac{T_{\text{ref}} + 273.15}{\bar{t} + 273.15} \tag{D1}$$

where q is the volume flow expressed at the reference density (the program works in mass, $\dot{m} = \rho_{\text{ref}}\,q$, so that mass balances at every junction however hot the air), $A\,L$ the airway's volume, and $\bar{t}$ the volume-weighted mean temperature of what it holds. **Hot air is thin**: the same mass of air at 300 °C takes up twice the volume, so an airway holds less of it and it moves faster. The second term in Eq. D1 is that effect: what no longer fits leaves at once. New air enters at the upstream end as a parcel carrying the upstream junction's state. Neighbouring parcels are merged (nearest in temperature and fume first) to at most 16 per airway, so the cost stays bounded. If the flow reverses, the same parcels simply travel the other way.

Plug flow has no diffusion. A front passes through an airway in exactly $A\,L/q$ (at the speed the heat gives the air), which is the check of Section D9.

D3. Mixing at junctions

The parcels that leave airways into a junction mix perfectly, weighted by their volume (at the reference state):

$$t_n = \frac{\sum_e v_e\, t_e}{\sum_e v_e}, \qquad \phi_n = \frac{\sum_e v_e\, \phi_e}{\sum_e v_e}, \qquad c_{n,k} = \frac{\sum_e v_e\, c_{e,k}}{\sum_e v_e} \tag{D2}$$

A junction that nothing flows into keeps its air. Intake junctions, and any junction the user marks as open to the surface, hold surface air: clean, at their own temperature. A very short airway, whose whole contents cross in one step, passes the rest straight through carrying what its upstream junction had; a link under 2 m long therefore makes fronts through it approximate, and the pre-run check says so.

D4. The fire

A fire sits in one airway and acts where the air leaves it, which is where the reference programs report it. It has a heat output and makes fume. Three types are modelled, after the classic definitions, with q the airflow reaching the fire and ϕ the fume already in that air:

- **Fixed heat:** the user gives the heat $\dot{Q}$ (kW) and the fume $\dot{V}_f$ (m³/s). Both follow the growth curve $\gamma(t)$. This is the type the program recommends, and the only one whose results the program stands behind.
- **Oxygen-rich:** the fire burns until a stated oxygen percentage is left downstream. The oxygen it uses is $q\,(21 - O_{2,\text{left}})/100$ m³/s, and the heat is that times 16 283 kJ per m³ of oxygen.
- **Fuel-rich:** all the oxygen burns; heat and fume are stated per m³ of oxygen: $\dot{Q} = (21/100)\, q \times k_{O_2}$.

The oxygen-limited types are marked not validated in the program: they reproduce the classic program's behaviour, which is itself unphysical in some of its own example cases (one of them asks for 134 % of the heat the air could stoichiometrically feed). **With a reference flow** Q_{ref}, the fire's heat follows the airflow, within half to double:

$$\text{follow} = 1 + \frac{q - Q_{\text{ref}}}{d}, \qquad d = \begin{cases} q + Q_{\text{ref}} & q < Q_{\text{ref}} \\ q & q \geq Q_{\text{ref}} \end{cases} \tag{D3}$$

For a fixed fire the heat is multiplied by follow. For an oxygen-rich fire the oxygen target moves with it, $O_{2,\text{left}} + (21 - O_{2,\text{left}})(\text{follow} - 1)$, and the oxygen use builds up over a lead time; for a fuel-rich fire the oxygen left falls to zero over the lead time, and the heat is divided by $\max(1,\ \text{follow})$.

A fixed fire cannot burn more than the air reaching it can feed. Air carries at most about 3.0 MJ per kg of air (13.1 MJ per kg of oxygen), so the heat is held to

$$\dot{Q} \leq 3030\, \rho_{\text{ref}}\, q \ \ \text{kW} \tag{D4}$$

and the program records in its log when the limit acts. Before this limit existed a collapsing airflow drove the fire's exit temperature to 284 923 °C.

Temperature leaving the fire

The air leaving the fire has taken up the fire's heat. The specific heat of air rises with temperature, $c_p = 0.2376 + 2.4 \times 10^{-5}\,T$ BTU/lb·°F with T in °F (the classic programs' units, which the code keeps), so the heat balance $\dot{m}\int c_p\,dT = \dot{Q}$ has a closed solution:

$$(T_{\text{out}} + 9900)^2 = (T_{\text{in}} + 9900)^2 + \frac{2\,\dot{Q}}{\dot{m}\,c_1}, \qquad c_1 = 2.4 \times 10^{-5}\ \text{BTU/(lb}\,\text{°F}^2\text{)} \tag{D5}$$

(9900 is $0.2376/2.4 \times 10^{-5}$.) $\dot{Q}$ is in BTU/min and $\dot{m}$ in lb/min, with the airflow converted at the reference density. The fume leaving is the fume in plus the fume made, over the volume in plus the volume the fire adds:

$$\phi_{\text{out}} = 100\,\frac{(\phi_{\text{in}}/100)\,q + \dot{V}_f}{q + \dot{V}_{\text{added}}} \tag{D6}$$

Gases follow from the fuel's *yields*, in kg per MJ of heat: oxygen 1/13.1 and carbon dioxide 0.087, measured at Runehamar across several fuels; carbon monoxide and smoke vary far more, and the program offers values for a mine vehicle, a diesel pool, a conveyor belt and mixed goods, each with its source. The tracer source of a fire is $c = y\,(\dot{Q}/1000)/q$, in kg per m³ of air at the reference state, and is shown as ppm, % or mg/m³ by $\text{ppm} = 10^6\,c/(M_w/22.414 \times 273.15/T_{\text{ref}})$.

Growth and sources

The fire's size over time is a factor $\gamma(t)$ of its full size: constant, a measured curve, or the standard t^2 growth, $\dot{Q} = \alpha\,t^2$ with α = 0.00293, 0.01172, 0.0469 or 0.1876 kW/s² (slow, medium, fast, ultrafast), up to the full size and then constant for a stated burn time, and then falling away over a minute. Gas and dust sources, other than fires, add kg/s of a tracer to an airway for a stated time. Where the air hardly moves ($|q| < 0.01$ m³/s) a source fills the airway instead of a near-zero flow carrying it off at a huge concentration.

Dust and smoke settle. A particle of diameter d and density ρ_p falls at the Stokes speed $v_s = \rho_p\,g\,d^2/(18\mu)$ ($\mu = 1.81 \times 10^{-5}$ Pa·s). Taken as well mixed across the airway, each step keeps the share $\exp(-v_s\,\Delta t/H)$ with H the airway's height, and the rest is added to what has settled out, reported per airway in kg.

D5. Heat lost to the walls

Two wall models are offered, because the classic one failed the only full-scale measurements the program has been tested on (Section D10).

The classic network rule (to compare with other codes)

The air in an airway cools or warms towards the rock as $t \leftarrow t_{\text{rock}} + (t - t_{\text{rock}})\, e^{-E\,\Delta x/c_p}$, with the exponent E from a film coefficient and Fourier conduction into rock that has been exposed since the airway was last disturbed. The film coefficient comes from a Reynolds-number correlation with a roughness correction from the airway's own friction factor, and the conduction from the series (small x) or asymptotic (large x) solution of conduction into a half-space with a convective surface, $F(x)$ with $x = (0.375 + \text{Bi})\sqrt{\tau}$, so that $\phi = \text{Bi} - F\,\text{Bi}^2/(0.375 + \text{Bi})$ and $E = k\,P^2\phi/(120\,\rho\,Q\,A)$ in the rule's own imperial units. The rock never warms: it keeps its starting temperature however long the fire burns. Against measured tunnel fires it read far too hot.

The tested model

The model checked against the Runehamar tests replaces it with three physical parts. **A film from the airway's own friction**, by the Reynolds analogy (Stanton number k/ρ for an Atkinson k), plus **radiation from the gas**:

$$h = \frac{\dot{m}}{A}\, c_p\, \frac{k}{\rho_{\text{ref}}} + \varepsilon\,\sigma\,(T_g^2 + T_w^2)(T_g + T_w) \tag{D7}$$

with ε an effective gas emissivity (0.3, fitted on one test, Section D10), $\sigma = 5.67 \times 10^{-8}$ W/m²K⁴ and absolute temperatures. The air in the airway cools towards the wall's surface temperature T_s while it is in the airway,

$$t \leftarrow T_s + (t - T_s)\,\exp\!\left(-\frac{h\,P\;\min(\Delta t,\;\tau_{\text{res}})}{\rho_g\,A\,c_p}\right), \qquad \tau_{\text{res}} = \frac{A\,L\,\rho_g}{\dot{m}} \tag{D8}$$

where τ_{res} is the time the air actually spends in the airway: a short link is cooled for its residence, not for a whole step (before this, a 5 m link at 5 m/s was over-cooled about five times). Third, **rock that warms and gives its heat back**: each airway has its own one-dimensional wall of ten cells, thin at the surface (first cell $\max(0.01, \sqrt{2.5\,\alpha\,\Delta t})$ m) and each 1.5 times thicker than the last into the rock. It starts as an error-function profile for the age the walls have already been exposed, and each step takes the surface flux $h\,(t_{\text{air}} - T_s)$ in, with explicit conduction between cells (sub-stepped to stay stable); the deepest cell holds its starting value. That memory matters most in the decay: at the same emissivity, the worst error at +458 m was about −80 °C without it and −23 °C with it.

Autocompression over the height the air travels in a step is shared between the two ends, as in Part C ($g\,\Delta z/c_p$); a link through surface air touches no rock and gets only that. Walls keep their settled age until the fire's air reaches them, and their conduction restarts from that moment.

D6. The airflow answers the fire

Every 15 s (30 s in the program's dialogue) the airflow is solved again. Each airway's air density comes from its own temperature, the parcels' volume-weighted mean plus the fire's

temperature rise in the fire's own airway (the whole airway is taken hot, as the classic programs treat it), at the reference pressure:

$$\rho_e = \rho_{\text{ref}} \frac{T_{\text{ref}} + 273.15}{t_e + 273.15} \tag{D9}$$

This is what makes throttling and natural draft. Hot air in a rise lightens the column, a fire in a flat drift throttles the flow because the resistance of an airway follows density, $R \propto \rho$ at a given volume flow (this applies to a resistance the user has entered as well, which the airflow solver would otherwise hold fixed). The airflow solver is the fire's own copy of Part C's, with friction $\Delta P = R\, q \sqrt{q^2 + q_L^2}$ with q_L the flow at Reynolds number 2000 in that airway, a square law at real flows and a straight line through zero for air barely moving (Section C5). Flows are then handed to the heat and fume as $\dot{m}/\rho_{\text{ref}}$.

Events. A fan can stop and the resistance of an airway can be multiplied (a door closed is a large factor, one opened its inverse) at stated times; the airflow answers at once. Before the fire starts the model runs **30 minutes of ventilation** by default so temperatures and flows settle.

D7. Reading a run

The program reads the result the way an engineer would: when the fire reached its peak, which junctions pass the limits for carbon monoxide (75 ppm), oxygen (19.5 %) and temperature (60 °C) and when, which airways reverse, how many airways have fume in them over time, how far the flow past the fire fell, and the **first arrival** of fume or a gas at every airway. Before a run, a **check** lists what the model cannot represent in this network (fixed-flow airways that cannot throttle or reverse, links under 2 m, airways with no section, no fan, a fire larger than its air can feed, oxygen-limited fire types), and after it the run can be compared with the steady airflow solve: before the fire they should agree, and a large gap at the start means the fire network differs from the solved one.

D8. Smoke upstream: the critical velocity

Smoke runs *against* the air along the roof when the air is slower than the critical velocity. The program uses the correlation of Li and others (2010), which agrees with Runehamar's 2.45 m/s at 6 MW:

$$v_c = \sqrt{g\,H} \times \begin{cases} 0.81\, q^{*1/3} & q^* \le 0.15 \\ 0.43 & q^* > 0.15 \end{cases}, \qquad q^* = \frac{\dot{Q}}{\rho\, c_p\, T\, \sqrt{g}\, H^{5/2}} \tag{D10}$$

with H the airway height. Where the air in the fire's airway is slower than v_c the program warns that smoke is likely to run upstream, and that this model shows clean air there: it has no upstream layer. Runehamar measured 15 to 25 m of it at 6 MW and about 100 m at 66 to 202 MW. Correlations for v_c disagree with one another, so the figure is an indication, not a limit.

D9. Worked example — a 5 MW fire in a drift

Three drifts in series, each 5 m × 5 m with an Atkinson $k = 0.012$: T1 (100 m) from the intake A at 101 365 Pa to B, T2 (200 m) from B to C, and T3 (300 m) from C to the exhaust D at 101 325 Pa, all at 20 °C. Before any fire the airflow carries 65.88 m³/s. A fixed fire of 5 000 kW making 0.3 m³/s of fume starts at once in T2 (Figure and tables below, tested wall model, time step 5 s, 10 minutes of settling first).

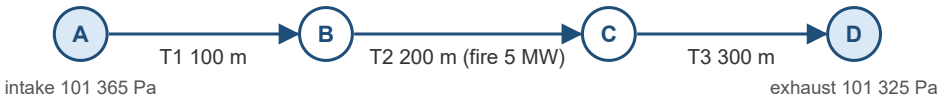


Figure 10. The fire example: three 5 m × 5 m drifts in series between an intake and an exhaust at the same elevation. The fire is in T2 and acts on the air leaving it, at C. Shaded: fixed-pressure nodes.

Hand check of the fire's exit temperature (Eq. D5)

The fire acts on the air that reaches it in the step it starts, which is the pre-fire flow. In the classic units, $T_{\text{in}} = 68$ °F, $\dot{Q} = 5000/0.01758 = 284414$ BTU/min, and $\dot{m} = 139594\ \text{cfm} \times 0.07491\ \text{lb/ft}^3$, so

$$(T_{\text{out}} + 9900)^2 = 9968^2 + \frac{2 \times 284414}{139594 \times 2.4 \times 10^{-5} \times 0.07491} = 101627468$$

giving $T_{\text{out}} = 181.0$ °F, **82.8 °C**. The program's first report shows 82.8 °C at the end of T2. (A fire of 5 MW in 66 m³/s is a rise of about 63 K.)

The flow answers the fire

Table 39. The 5 MW fire in T2: flows and temperatures with the two wall models. The flow falls because the hot airway's resistance rises with the thinner air (Eq. D9).

Quantity	Before fire	t = 0 (first report)	t = 1800 s, tested wall	t = 1800 s, classic rule
Flow in T2 (m³/s)	65.88	63.52	62.11	61.25
Fall from pre-fire flow (%)	—	3.6	5.7	7.0
Air leaving T2 (°C)	20.0	82.8	86.6	87.5
Air leaving T3 (°C)	20.0	20.0	35.0	50.9
Fume at D (%)	0	0.000	0.481	0.487

The flow falls by 5.7 % with the tested wall and 7.0 % with the classic rule: both throttle, the classic rule more because it leaves the air in T2 hotter (87.5 against 86.6 °C). Downstream the difference is large: after 30 minutes the air leaving T3 is 35.0 °C with the tested wall,

where the rock has warmed and keeps taking heat, and 50.9 °C with the classic rule, which loses it faster to a rock that never warms.

Plug-flow arrival (Section D2)

The fire's fume leaves T2 at $t = 0$. It has to cross T3 ($A\,L = 25 \times 300 = 7500$ m³) to reach D. At 63.5 m³/s (at the reference density) and the air's temperature, which makes the air in T3 about 1.21 times its reference volume, the crossing takes $7500/(63.5 \times 1.21) = 97$ s. In the program fume is first reported at D in the report at 120 s (the reports are 30 s apart); the report before it shows none.

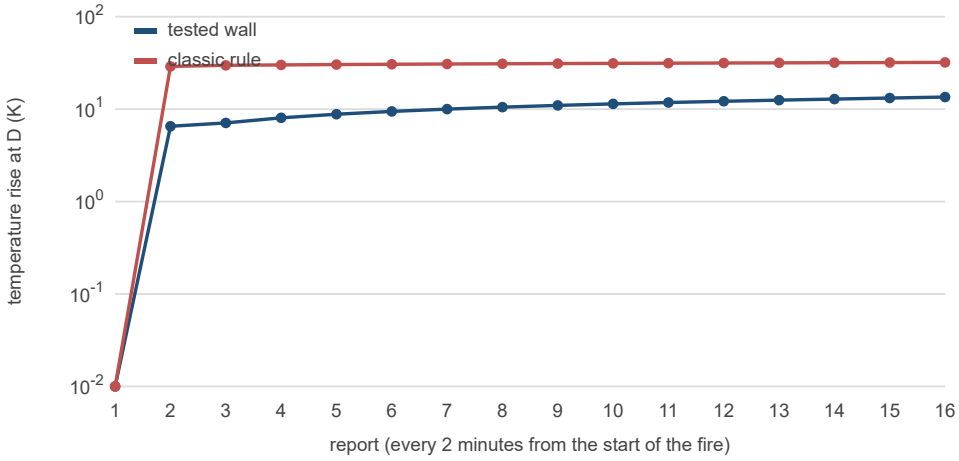


Figure 11. The 5 MW fire: temperature rise of the air at the exhaust D over the 30 minutes. The front arrives after about two minutes; the classic wall rule leaves the air about twice as hot.

D10. How far can it be trusted?

Against measurement. The tested wall model was checked against the Runehamar tunnel fire tests (SP Technical Research Institute of Sweden, report 2011:55): a 1 600 m road tunnel with fires of 6 to 202 MW. Air temperature 458 m from the fire:

Table 40. Peak air temperature at +458 m in the Runehamar tests against the two wall models (the measured velocity imposed, so this tests heat and transport alone). One value was fitted, the gas emissivity, on T4 only; T0, T1, T2 and T3 are hold-outs.

Test	Peak heat (MW)	Measured peak (°C)	Tested wall (°C)	Classic rule (°C)
T0 (hold-out)	6.1	22.1	18.4	31.4
T4 (fitted)	66.1	72.8	72.0	230.9
T3 (hold-out)	117.1	78.0	90.2	381.5
T2 (hold-out)	155.1	93.3	106.3	481.9
T1 (hold-out)	201.2	133.2	131.8	622.9

As a share of the rise above ambient the peak errors are −33.0, −1.3, +17.5, +15.6 and −1.2 % (T0, T4, T3, T2, T1); in degrees, −3.7 to +13.0 °C. The classic rule reads 31 to 623 °C where 22 to 133 °C was measured. The velocity predicted from the fans' thrust falls within −15 to +7 % of measured, about as well as the test authors' own correlation, which supports

keeping resistance proportional to density. Arrival at +458 m is 30 to 40 s late on a trip of 3 to 4 minutes. The inputs alone (emissivity, rock conductivity, perimeter, the heat release rate's own ±15 %) move the peak by 15 to 40 %, so the agreement should not be read as tighter than that.

What this does not show. It is one straight road tunnel at 2 to 2.5 m/s, not a mine network. It does not test reversal, branching, low-velocity drifts or rough rock. Oxygen is not an independent check, because the heat release rate was derived from it. Near the fire, a section mean cannot be compared with thermocouples under the ceiling: against those the model is −51 to +33 % of the ceiling rise at 100 m. The model has no ceiling layer, no backlayering and no carbon monoxide yield of its own. The emissivity is one number standing for everything between gas and rock in a large smoky tunnel, fitted on one tunnel. **Fire size** is the largest uncertainty of all: an estimate of the heat release of a vehicle can be wrong by a factor of 2 to 6, and the carbon monoxide yield by 2 to 20, which dwarf every figure above once the inputs are not exact.

Against the classic programs. Where the classic wall rule is selected, the model reproduces the settled flows and temperatures of the classic program's example networks to within a few per cent on most of them, and not their timing or throttling: on two large grid networks the classic program shows fume in about twice as many airways at 2 000 s as plug flow can have reached, which a hand calculation of plug-flow arrival confirms. Agreement between two codes is a consistency check, not accuracy, and neither is tested in a mine at the fire sizes a mine is designed for. The defensible statement for a design report is that the flow in main airways is good to about 10 to 20 %, and that flow in small branches, temperature near the fire, smoke upstream and the figures for fume and carbon monoxide are not predictions: use them to compare options.

Part E — The CFD solver

E1. What it is for

The network solvers of Parts A and C treat an airway as one number, its resistance. The CFD solver looks *inside* a tunnel: the speed and pressure of the air at every point of its section, how the flow develops after a bend, a change of opening or a duct, and what pressure drop a stretch of tunnel really gives. It solves the **steady incompressible flow** of air on a three-dimensional mesh that follows the tunnel's own walls, and returns the pressure drop, the Atkinson factor and resistance that reproduce it, and the fields for the viewer (speed, streamlines, wall shear).

It is not coupled to the heat. Density is constant, and temperature plays no part. It is also, in the words of its own source, the *first working solver*: it conserves mass exactly and converges, and its pressure drop is checked against the friction law of Colebrook and White (Section E8), but it has not yet been calibrated against a resolved reference such as OpenFOAM on the same mesh, and its junction model is a lumped estimate (Section E3). The mesh can be written as an OpenFOAM `polyMesh` (points, faces, owner, neighbour, boundary) so that exactly the same cells can be solved by both.

E2. The mesh: the section swept along the path

A voxel mesh would give a tunnel a stepped wall. The CFD mesh instead follows the tunnel's own surface. The section outline (a rectangle, an arch, a drawn profile, or the outline cut from a GLB model of the tunnel at each station) is swept along the centre line, with the path running through the point on the section where the node sits. A tunnel whose opening changes along its length (a taper) blends the outline station by station.

The O-grid. The outline is cut into segments, keeping its corners. Rings run from the wall to a small core: between two rings each segment makes a quadrilateral, which swept along the path becomes a hexahedron; the core is one polygon, which becomes a prism with M sides. The result is a general polyhedral finite-volume mesh in OpenFOAM's layout: a list of faces, each with an owner cell, and, for the internal ones, a neighbour.

Wall layers. Thin cells at the wall grow towards the middle. The ring thicknesses, measured as fractions of the radius from the wall (1) to the core ($s_{\text{core}} = 0.15$), start at a first layer y_1 and grow geometrically,

$$h_{i+1} = r\,h_i, \qquad r = 1.25 \text{ (at least } 1.05), \qquad h_i \le \max(y_1,\ \text{ring size}) \tag{E1}$$

until the core share is reached. The first layer is chosen so that the wall cell is in the logarithmic layer, where the wall function (Section E5) is valid: the program takes a friction speed $u_\tau \approx 0.045\,U_{\text{in}}$, wants a wall distance of $y^+ \approx 200$, and for a rough wall of at least $1.5\,k_s$, then takes twice that, within a window of 0.002 to 0.12 of the hydraulic diameter:

$$y_1 = \min(\,\max(2\max(\frac{200\,\nu}{0.045\,U_{\text{in}}},\ 1.5\,k_s),\ 0.002\,D_h),\ 0.12\,D_h) \tag{E2}$$

The wanted number of segments and stations depends on the fineness chosen: coarse 28 × 30, medium 44 × 50, fine 64 × 80 (segments around the section × stations along the tunnel). When the first layer cannot be kept above the roughness the program says so in its notes.

Several tunnels in one mesh

Every tunnel is meshed along its own path and section. Where two or more tunnels meet at a node, each is trimmed back to the point of contact and the end faces of all of them join through **one lumped junction cell**: the end faces become internal faces between the tunnel's last cells and the junction cell, so there is no wall disc at a port and no overlap, and the whole network is one conforming mesh. A node where only one tunnel ends stays a boundary: an inlet, with its speed from the network solve ($U = Q/A$), or an outlet. A hull of triangles round the tunnel ends forms the junction's wall, leaving each end open.

The limit, stated plainly. The junction is one cell, so the flow inside it is not resolved: the solver sees the pressure and the momentum exchanged through its faces, but not the corner vortices or the separation at a sharp T. The mesh conserves mass; the junction loss it gives is a lumped estimate that has to be calibrated against a resolved junction before it is trusted.

E3. The equations

The flow is steady, incompressible and turbulent. With the Reynolds-averaged equations, the velocity $\mathbf{U}$ and the kinematic pressure $p = P/\rho$ satisfy

$$\nabla\cdot\mathbf{U} = 0, \qquad \nabla\cdot(\mathbf{U}\,\mathbf{U}) = -\nabla p + \nabla\cdot[(\nu + \nu_t)\,(\nabla\mathbf{U} + \nabla\mathbf{U}^T)] \tag{E3}$$

where $\nu = 1.5 \times 10^{-5}$ m²/s is the kinematic viscosity of the air and ν_t the eddy viscosity from the turbulence model (Section E5). The pressure is kept kinematic inside the solver and given in Pa (multiplied by $\rho = 1.2$) in the results, with the outlet as the gauge reference. There is no gravity, no buoyancy and no temperature.

Boundaries. At an inlet the velocity is imposed, uniform and normal to the face at the stated speed (the mean speed of the airway, or, for a duct opening into a tunnel through a bulkhead, the speed of the jet through the opening, with the rest of the inlet closed). At an outlet the gauge pressure is zero and the velocity gradient is zero. At a wall the velocity is zero, applied through the wall function.

E4. The finite-volume method

Each cell is a polyhedron with a volume V and a centre; each face f has an area vector $\mathbf{S}_f$. Integrating Eq. E3 over a cell and applying Gauss's theorem turns the divergences into sums over the faces. The program uses:

- **Gradients** by Gauss's theorem, $(\nabla\phi)_c = \frac{1}{V_c}\sum_f \phi_f \mathbf{S}_f$, with face values interpolated linearly by the weight $w_f = |\mathbf{S}\cdot(\mathbf{C}_N - \mathbf{F})|/(|\mathbf{S}\cdot(\mathbf{F} - \mathbf{C}_P)| + |\mathbf{S}\cdot(\mathbf{C}_N - \mathbf{F})|)$.
- **Diffusion** through each internal face as $D_f = (\nu + \nu_{t,f})\,|\mathbf{S}_f|^2/(\mathbf{S}_f\cdot\mathbf{d})$, with $\mathbf{d}$ the vector between the two cell centres.
- **Convection** by first-order upwind, so the coefficient to a neighbour is $-(D_f + \max(\mp F_f, 0))$ with F_f the volume flux through the face.
- **Under-relaxation** of the momentum equations ($\alpha_U = 0.7$) and of the pressure ($\alpha_P = 0.3$).

SIMPLE with Rhie–Chow fluxes. One iteration solves the momentum equations for a velocity $\mathbf{U}^*$ with the latest pressure (four Gauss–Seidel sweeps), forms the face fluxes with the Rhie–Chow correction so that velocity and pressure do not decouple on a collocated mesh,

$$F_f^* = \overline{\mathbf{U}^*}_f\cdot\mathbf{S}_f - \overline{(V/a_P)}_f\left[\frac{|\mathbf{S}_f|^2}{\mathbf{S}_f\cdot\mathbf{d}}(p_N - p_P) - \overline{\nabla p}_f\cdot\mathbf{S}_f\right] \tag{E4}$$

(the compact face pressure gradient less the interpolated one), and then solves a pressure-correction equation for p' that makes the corrected fluxes conserve mass in every cell,

$$\sum_f c_f\,(p'_N - p'_P) = -\sum_f F_f^*, \qquad c_f = \overline{(V/a_P)}_f\,\frac{|\mathbf{S}_f|^2}{\mathbf{S}_f\cdot\mathbf{d}} \tag{E5}$$

by preconditioned conjugate gradients (diagonal preconditioner, up to 600 iterations or until the residual has fallen by 10^{-4}). Then $F_f \leftarrow F_f^* - c_f\,(p'_N - p'_P)$, $\mathbf{U} \leftarrow \mathbf{U}^* - (V/a_P)\nabla p'$ and $p \leftarrow p + \alpha_P\,p'$. The iteration stops when the continuity imbalance, summed over all cells and taken as a share of the inlet flow, is below 10^{-5} (10^{-6} in the tunnel runs). It is reported with the momentum residual, scaled by the flow.

The junction cell. In a network the junction cell has no shape of its own: its volume and centre are given. Air leaving a tunnel carries its own momentum into the junction (upwind), and air entering a tunnel from the junction comes in along the face normal at the face speed, because a lumped cell has no direction of its own.

E5. Turbulence and the wall

Two models for ν_t are offered. The default is the standard k–ε model; the other is a mixing length.

Standard k–epsilon

The eddy viscosity is $\nu_t = C_\mu\,k^2/\varepsilon$, with transport equations for the turbulent kinetic energy k and its dissipation ε:

$$\nabla\cdot(\mathbf{U}\,k) = \nabla\cdot\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\nabla k\right] + P_k - \varepsilon, \qquad \nabla\cdot(\mathbf{U}\,\varepsilon) = \nabla\cdot\left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon}\right)\nabla\varepsilon\right] + C_{1\varepsilon}\frac{\varepsilon}{k}P_k - C_{2\varepsilon}\frac{\varepsilon^2}{k} \tag{E6}$$

with the usual constants $C_\mu = 0.09$, $C_{1\varepsilon} = 1.44$, $C_{2\varepsilon} = 1.92$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$, and the production $P_k = \min(\nu_t S^2,\ 50\,\varepsilon)$ with $S^2 = 2\,S_{ij}S_{ij}$; the limit stops the production running away in stagnation regions. Convection is upwind, diffusion as in Section E4, the equations are under-relaxed by 0.8, and k and ε are kept positive. The inflow takes a turbulence intensity I (5 % of the inlet speed by default) and a length $\ell = 0.07\sqrt{4A_{in}/\pi}$:

$$k_{in} = \tfrac{3}{2}\,(I\,U_{in})^2, \qquad \varepsilon_{in} = \frac{C_\mu^{3/4}\,k_{in}^{3/2}}{\ell} \tag{E7}$$

Mixing length

A simpler, cheaper model takes the eddy viscosity from the strain and a length ℓ_m that depends on the distance y from the wall through Nikuradse's profile, with R the largest wall distance in the mesh:

$$\nu_t = \ell_m^2\sqrt{S^2}, \qquad \ell_m = R[0.14 - 0.08\,(1-y/R)^2 - 0.06\,(1-y/R)^4] \tag{E8}$$

The wall function

The mesh does not resolve the viscous sublayer. The shear at the wall comes from a law of the wall, solved for the friction speed u_τ in each wall cell by iteration (30 damped steps), from the speed u_p of the cell next to the wall at its distance y. For a smooth wall,

$$\frac{u_p}{u_\tau} = \frac{1}{\kappa}\ln(E\,y^+), \qquad y^+ = \frac{y\,u_\tau}{\nu}, \qquad \kappa = 0.41,\ E = 9.8 \tag{E9}$$

For a rough wall with equivalent sand roughness k_s and $k_s^+ = k_s u_\tau/\nu$: when $k_s^+ \le 2.25$ the wall is smooth; between 2.25 and 90 it is *transitionally rough*, the smooth law shifted down by $\Delta B = \frac{1}{\kappa}\ln(1 + 0.3\,k_s^+)$; above 90 it is *fully rough*,

$$\frac{u_p}{u_\tau} = \frac{1}{\kappa}\ln\!\left(\frac{30\,y}{k_s}\right) \tag{E10}$$

The effective wall viscosity is then $\nu_w = u_\tau^2\,y/u_p$ (so that the shear is $\nu_w u_p/y$), the eddy viscosity in the wall cell is at least $\kappa\,u_\tau\,y$, and with k–ε the wall cell is held at $k = u_\tau^2/\sqrt{C_\mu}$ and $\varepsilon = u_\tau^3/(\kappa\,y)$. The program reports the range of y^+ over the wall; it means something only for a smooth wall (for a rough wall y^+ is large by construction).

E6. Roughness: from Atkinson's k

The network solvers describe a tunnel's roughness by Atkinson's friction factor k (kg/m³). The CFD needs an equivalent sand roughness k_s (m). They are linked through the Darcy friction factor, $\lambda = 8k/\rho$, and the Colebrook–White law,

$$\frac{1}{\sqrt{\lambda}} = -2\log_{10}\!\left(\frac{k_s/D_h}{3.7} + \frac{2.51}{Re\sqrt{\lambda}}\right), \qquad Re = \frac{U\,D_h}{\nu} \tag{E11}$$

The program finds k_s by bisection so that the Colebrook factor at the run's Reynolds number equals $8k/\rho$. A tunnel smoother than smooth walls can give (target below the smooth value)

is smooth. A very rough tunnel gives a large k_s: for $k = 0.012$ in a 4 m tunnel at 3 m/s it is a quarter of a metre, 6 % of the diameter, which is more roughness than a wall function can honestly describe (the program notes it when the first cell cannot be kept above it).

E7. What comes out

From the converged field the program reports the mass imbalance, the flow leaving (which should equal the flow entering to the tolerance), the largest and smallest y^+, and for a tunnel the **pressure drop along the developed part** $\mathrm{d}P/\mathrm{d}x$ (Pa/m), from the slope of the cross-section mean pressure between two stations away from the ends, and over the whole run. The two numbers a network model can take from it:

$$k_{\mathrm{eq}} = \frac{(\mathrm{d}P/\mathrm{d}x)\,A}{\mathrm{Per}\,U^2}, \qquad R = \frac{(\mathrm{d}P/\mathrm{d}x)\,L}{Q^2} \tag{E12}$$

the Atkinson factor and the resistance that reproduce the CFD's pressure drop. For a tapering tunnel it also reports the ideal pressure recovery $\frac{1}{2}\rho(V_1^2 - V_2^2)$ and how much of it the flow regains; for a duct jet, the centre-line decay against $U\min(1,\ 5.8\,d/x)$ and the share of the volume in reverse flow.

E8. Worked example — a straight tunnel

A straight 60 m tunnel 4 m × 4 m ($D_h = 4$ m), air at 3 m/s ($Q = 48$ m³/s, $Re = 8.00 \times 10^5$), at the reference density 1.2 kg/m³. Colebrook–White for a smooth wall gives a Darcy factor $\lambda = 0.01210$ and a gradient $\lambda/D_h \times \frac{1}{2}\rho U^2 = 0.01634$ Pa/m. With the roughness of an Atkinson $k = 0.012$ ($k_s = 0.252$ m), $\lambda = 8k/\rho = 0.08000$ and the gradient is 0.1080 Pa/m.

Table 41. The straight 4 m tunnel: smooth and rough walls on two mesh sizes (4 m × 4 m, 60 m, 3 m/s, k–epsilon). Every run converged to the mass tolerance.

Case	Mesh (segments × rings × stations)	Cells	First layer (m)	Iterations	Mass imbalance	y^+ range	dP/dx (Pa/m)	k_{eq} (kg/m³)
smooth coarse	28 × 12 × 31	10110	0.0444	67	9.14×10^{-7}	146 – 191	0.0176	0.00195
smooth medium	44 × 12 × 51	26450	0.0444	118	9.95×10^{-7}	147 – 196	0.0176	0.00196
rough coarse	28 × 4 × 31	3390	0.4610	61	7.83×10^{-7}	3612 – 4759	0.0973	0.01081
rough medium	44 × 5 × 51	11050	0.4599	100	9.88×10^{-7}	3643 – 4868	0.0987	0.01097

Smooth wall. The CFD pressure gradient is 0.0176 Pa/m on the coarse mesh and 0.0176 Pa/m on the medium, against 0.0163 by Colebrook–White: 7.7 % high, and the same on both meshes, so it is not a mesh error. It is what the wall function and the k–ε model give. The first cell sits at y^+ of 147 to 196, inside the logarithmic layer as intended.

Rough wall. With the roughness of $k = 0.012$ the CFD gradient is 0.0973 (coarse) and 0.0987 (medium) Pa/m against 0.1080: −8.6 % on the medium mesh. Fed back to the network model, the CFD says $k_{\mathrm{eq}} = 0.0110$ for an input of 0.012, −8.6 %. The program warns that this roughness is large for the tunnel, and the result is a little low on both meshes; the first wall cell of 0.46 m is as thick as the roughness allows.

What this example shows. The solver converges to the mass tolerance, conserves mass (the flow leaving is the flow entering to 5.89×10^{-6} %), is insensitive to refining the mesh in its pressure drop, and gives the friction law's pressure drop to within about 10 %, high for a smooth wall and low for a very rough one. A calibration against OpenFOAM on the same mesh and a rough-wall formulation that does not need the first cell to be thicker than the roughness are still to be done, and until then a CFD pressure drop is an estimate to about 10 %, not a measurement.

A T junction

Three 4 m × 4 m tunnels of 12 m meet at one junction: A brings 32.0 m³/s, and the network solve says B takes 20 and C takes 12. Both outlets are at the same (zero gauge) pressure and are not told those numbers: the CFD shares the air itself. The mesh has 4339 cells; the solve converged in 425 iterations. The flow leaving is 32.00 m³/s against 32.00 m³/s entering (9.20×10^{-5} % apart), and the openings are:

Table 42. The T junction: the flow at every opening.

Node	Tunnel	Role	Flow (m³/s)
a	A	inlet	32.00
b	B	outlet	20.06
c	C	outlet	11.94

This shows the mesh and the mass balance, not the junction loss: with one lumped cell for the junction, the pressure loss across it is an estimate, to be calibrated (Section E2).

References

Carslaw, H.S. and Jaeger, J.C. (1959). *Conduction of Heat in Solids*, 2nd ed. Oxford University Press (§13.5, the cylinder with a convective surface).

du Plessis, J.J.L. (ed.) (2014). *Ventilation and Occupational Environment Engineering in Mines*, 3rd ed. Mine Ventilation Society of South Africa (“the Bluebook”).

George, A. (1971). *Computer implementation of the finite element method*. PhD thesis, Stanford University (the reverse Cuthill–McKee ordering).

Cuthill, E. and McKee, J. (1969). Reducing the bandwidth of sparse symmetric matrices. *Proc. 24th ACM National Conference*, 157–172.

Goch, D.C. and Patterson, H.S. (1940). The heat flow into tunnels. *J. Chem. Metall. Min. Soc. S. Afr.* 41, 117–128.

Hartman, H.L., Mutmansky, J.M., Ramani, R.V. and Wang, Y.J. (1997). *Mine Ventilation and Air Conditioning*, 3rd ed. Wiley.

McPherson, M.J. (1993). *Subsurface Ventilation and Environmental Engineering*. Chapman & Hall (ch. 7 network analysis; ch. 14 psychrometry; ch. 15 heat in mines).

Meijerink, J.A. and van der Vorst, H.A. (1977). An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Mathematics of Computation* 31, 148–162.

PEE 410 (2018). *Mine Environmental Engineering* course notes (“the Blackbook”). University of Pretoria.

Saad, Y. and Schultz, M.H. (1986). GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* 7, 856–869.

Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*, 2nd ed. SIAM.

Starfield, A.M. (1966). Tables for the flow of heat into a rock tunnel with different surface heat transfer coefficients. *J. S. Afr. Inst. Min. Metall.* 66, 692–694.

Stehfest, H. (1970). Algorithm 368: numerical inversion of Laplace transforms. *Comm. ACM* 13, 47–49.