E3B Developer’s Theory Manual - M3

Double Tapered Composite Beam: Buckling Analysis

This manual documents the theory and implementation of the E3B local face-buckling solver for a double-tapered laminated box beam. The solver combines micromechanics, Classical Laminate Theory (CLT), a Rayleigh-Ritz buckling formulation, and a corrected Koiter-type post-buckling model. It also supports one-at-a-time (OAT) sensitivity analysis and a Saltelli-lite Sobol uncertainty quantification (UQ) workflow.

0. Document Control

Document	E3B Developer’s Theory Manual - M3
Tool names (GUI labels)	Double Tapered Composite Beam: Buckling Analysis (single-case) Double Tapered Composite Beam: OAT Sensitivity Analysis Double Tapered Composite Beam: Sobol Uncertainty Quantification
Scope version	M3 solver core (two-term Ritz + optional root rotational spring); M2 retained as a fast approximation.
Author / owner	[Fill in]
Version / date	v1.0 / 2026-01-02
Change log (high level)	M2a -> M3 core: two-term longitudinal Ritz basis and optional root rotational spring. FULL mode produces load-deflection curve, mode contour, and DeflectionGrid export. SENS and SOBOL respect the selected solver core (m2 or m3) for consistent comparisons. Corrected Koiter post-buckling uses face A-matrix and closed-form width integrals.

1. Purpose and Scope

1.1 What the solver predicts

The solver predicts the critical local face-buckling load (Pcr) of a thin-walled, laminated, double-tapered box beam subjected to a transverse tip load in a cantilever configuration. Local buckling is represented by a single sinusoidal half-wave across the interior face panel (between webs) and a Ritz basis along the span to capture root restraint and modal decay.

In FULL (single-case) mode, the solver additionally generates a Koiter-type post-buckling load-deflection curve and a normalized mode contour of the local face deflection.

1.2 Supported run modes

- FULL: Single-case evaluation (buckling + post-buckling curve + mode contour + deflection grid export).
- SENS: One-at-a-time (OAT) parameter sweep for Pcr around a baseline design point.
- SOBOL: Saltelli-lite Sobol UQ (first-order and total-order indices) using two base sampling matrices.

1.3 What is not modeled

The solver is assumption-driven and does not model the following effects:

- Global beam instabilities (overall Euler buckling, global lateral-torsional buckling).
- Progressive damage or delamination (interlaminar failure, ply cracking, debonding).
- Material nonlinearity, plasticity, viscoelasticity, or rate effects.
- Geometric nonlinearity beyond the simplified Koiter-type post-buckling amplitude (no full nonlinear equilibrium path following).
- Imperfection sensitivity (no explicit initial imperfection field; the post-buckling curve is theoretical).
- Local web buckling or flange local buckling (only the interior face panel width mode is modeled).

2. Geometry and Coordinate Definitions

2.1 Coordinate system

The beam is modeled as a cantilever with spanwise coordinate x measured from the clamped root (x = 0) to the tip (x = L). The local panel-width coordinate y spans the interior face between the two webs. In the formulation, y is defined on [0, b(x)], where b(x) is the local panel width. For visualization only, the contour plot is displayed as y centered about zero (i.e., y_plot = y - b(x)/2). The thickness direction z is normal to the laminate mid-surface and follows the standard CLT convention (z = 0 at the laminate mid-plane).

2.2 Primary geometry parameters

- L [m]: beam length.
- b_root, b_tip [m]: interior face-panel width at root and tip (linear taper).
- h_root, h_tip [m]: web height at root and tip (linear taper).
- w_f [m]: flange width (each side), modeled as a bonded strip at +/- H(x)/2.

2.3 Derived fields used by the solver

The solver forms spanwise distributions by linear interpolation between root and tip values:

- b(x) = b_root + (b_tip - b_root) * (x / L)
- h(x) = h_root + (h_tip - h_root) * (x / L)
- H(x) = h(x) + 2 * t_face_total (outer height including faces)
- k_y(x) = pi / b(x) (width-wise wavenumber of the assumed sinusoidal mode)

These fields are evaluated on a 1D grid along x. The local wavenumber k_y(x) is used consistently in both the buckling energy terms and the post-buckling (Koiter) membrane terms.

3. Material Modeling

3.1 Constituent inputs

The lamina properties are derived from fiber and matrix constituent inputs. The solver expects isotropic fiber and matrix properties as:

- Ef [Pa], Gf [Pa], nuf [-]: fiber Young's modulus, shear modulus, and Poisson's ratio.
- Em [Pa], (Gm is computed as Em / (2(1 + num))), num [-]: matrix Young's modulus and Poisson's ratio.
- Vf [-]: fiber volume fraction (Vm = 1 - Vf).

3.2 Micromechanics outputs

From the constituents, the solver computes an equivalent transversely isotropic UD lamina (E1, E2, G12, nu12). A simple mixture/empirical scheme is used: E1 uses a rule of mixtures, while E2 and G12 use a Halpin-Tsai-type form to capture transverse/shear efficiency. nu12 is computed by a linear mixture of the constituent Poisson's ratios.

These micromechanics outputs are intended to provide consistent lamina inputs for CLT when only constituent data are available. If calibrated lamina properties exist (from coupon tests), they should be used to back-calculate consistent constituent inputs or replaced upstream of the solver.

3.3 Reduced stiffness Q and transformed Q-bar

The lamina reduced stiffness matrix Q (plane stress) is formed from (E1, E2, G12, nu12). For each ply orientation theta, the solver computes the transformed reduced stiffness Q-bar using standard tensor transformation relations. Q-bar is then integrated through the thickness in the CLT step to form laminate A, B, and D matrices.

4. Classical Laminate Theory

4.1 Layup definition

Two independent laminates are defined: one for the faces and one for the webs. Each layup is provided as a comma-separated list of ply angles in degrees, ordered from bottom to top. The total laminate thickness is specified separately for face and web; the solver assumes uniform ply thickness within each laminate.

- face_angles [deg], t_face_total [m]
- web_angles [deg], t_web_total [m]

4.2 ABD computation

Given Q-bar for each ply and the through-thickness z-coordinates, the solver integrates Q-bar to form A, B, and D matrices. B is computed for completeness, but the current buckling and post-buckling implementation relies on A and D as described below.

4.3 How the laminate matrices are used

Different matrices enter different parts of the formulation and outputs:

- D_face: bending energy terms in the Rayleigh-Ritz buckling functional (dominant stiffness controlling Pcr).
- A_face: membrane stiffness used to compute the Koiter quartic coefficient (post-buckling amplitude growth).
- D_web: web bending stiffness used to model edge rotational restraint of the face panel.
- A_web: contributes to section EI for the linear tip-deflection estimate (delta_lin).

This separation allows the solver to capture the leading physical roles of each structural component while keeping the formulation computationally lightweight.

5. Load Mapping and Section Properties

5.1 Thin-wall mapping concept

The buckling functional is written in terms of an effective in-plane stress resultant induced in the face panel by the applied tip load. Instead of solving a full beam bending problem, the solver uses a thin-wall mapping: the tip load produces a spanwise bending moment M(x) = P (L - x), and the corresponding face axial resultant is proportional to the distance from the neutral axis and the section moment of inertia.

5.2 Section moment of inertia I_total(x)

The spanwise second moment of area I_total(x) is assembled from three contributions:

- Face panels: treated as thin strips located at +/- H(x)/2 (dominant in thin-walled bending).
- Webs: treated with a rectangular section inertia contribution proportional to t_web_total * h(x)^3.
- Flanges: treated as bonded build-ups at +/- H(x)/2 with effective thickness t_face_total + t_web_total and width w_f.

The flange contribution affects only bending stiffness and load mapping; flanges do not add edge-rotation restraint in the current model (see Section 7).

5.3 Definition and meaning of alpha0(x)

The mapping is expressed through a spanwise weighting alpha0(x), which scales the geometric/work term in the Rayleigh-Ritz formulation. Physically, alpha0(x) represents how strongly the external bending field couples into the local face-buckling mode at position x. It increases where the bending moment is high (near the root) and decreases toward the tip.

5.4 EI_total(x) for linear tip deflection

For the FULL mode load-deflection plot, the solver also computes a linear tip deflection component using an equivalent bending stiffness EI_total(x). EI_total(x) is assembled from face, web, and flange contributions using the laminate axial stiffness (A11) of each component. The linear compliance fb is obtained by integrating the standard cantilever kernel (L - x) x / EI_total(x) along the span.

6. Buckling Formulation (Rayleigh-Ritz)

6.1 Assumed displacement field

The local face out-of-plane deflection is assumed separable in x and y:

w(x, y) = F(x) * phi(y)

Across the panel width, the solver uses a single sinusoidal half-wave:

phi(y) = sin(pi y / b(x)), y in [0, b(x)]

6.2 Longitudinal basis functions

Along the span, a decaying exponential-trigonometric form is used to represent a root-influenced local mode that decays away from the root.

- M2 core (fast): one-term basis F2(x) = exp(-alpha x) * x * sin(beta x).
- M3 core (default): two-term basis F(x) = c1 F1(x) + c2 F2(x), where F1(x) = exp(-alpha x) * sin(beta x) and F2(x) = exp(-alpha x) * x * sin(beta x).

The two-term M3 basis improves flexibility near the root and provides a more robust representation when an explicit root rotational spring is included.

6.3 Energy structure and matrix form

The Rayleigh-Ritz method minimizes the total potential energy. After substituting the assumed mode shape, the problem reduces to evaluating x-integrals of bending energy (from D_face), coupling terms, shear-related terms, and an edge-rotation energy penalty. These contributions form a stiffness-like matrix N. The work/geometric term induced by the mapped bending field forms a matrix D.

- N: contains bending stiffness and edge-rotation restraint contributions.
- D: contains the geometric/work term constructed from alpha0(x) and F'(x).

6.4 Generalized eigenvalue problem

The critical load is obtained from the generalized eigenproblem:

N c = P * D c

For M2, c is a scalar and P is evaluated directly for each (alpha, beta). For M3, c is a 2-component vector and P is the smallest eigenvalue of D^{-1} N for each (alpha, beta) pair.

6.5 Search strategy for alpha and beta

The solver searches over the Ritz parameters (alpha, beta) to find the minimum critical load. A width-aware beta window is defined using the root panel width, and alpha bounds are adjusted based on an effective non-dimensional stiffness parameter. The search proceeds in two stages:

1. Coarse scan over a small grid in (alpha, beta) to locate a promising region.
2. Local refinement with a few iterations using narrower beta and alpha windows around the current best point.

This strategy provides a robust minimum while keeping computation time low for sensitivity and UQ runs.

7. Edge-Rotation Modeling

7.1 Web-only rotational restraint

Local face buckling between webs is influenced by rotational restraint at the panel edges. In this solver, the restraint is modeled as an equivalent rotational spring derived from the web bending stiffness about the face edge line. Each web provides a rotational stiffness proportional to D22_web / h(x), and the two webs are combined as a pair.

k_theta_web_pair(x) = 2 * (4 * D22_web / h(x))

7.2 Root rotational spring (M3 only)

To represent additional clamping or fixture effects at the root, the M3 core optionally includes an explicit root rotational spring per unit width, Ktheta_root_per_m. This term enters the N11 component (associated with the F1 basis) as an additional energy penalty proportional to beta^2 and the root half-width.

- Ktheta_root_per_m -> larger values approach a clamped-like root response.
- Smaller values allow more rotation and typically reduce Pcr.

7.3 Why flanges do not contribute to edge clamp

Flanges are treated as bonded build-ups that increase section bending stiffness and influence load mapping, but they are assumed free in y-rotation and therefore do not add edge rotational restraint. This assumption isolates the edge-clamping role to the webs, which is consistent with the intended 'interior panel between webs' local buckling model. If flange edge restraint is significant for a particular design, the current solver will under-predict the effective clamp and may under- or over-estimate Pcr depending on the configuration.

8. Post-buckling (Corrected Koiter)

8.1 Overview

Beyond the critical load, the solver uses a simplified Koiter-type post-buckling model to estimate how a local-buckling amplitude grows with load. The implementation is 'corrected' in the sense that the quartic membrane coefficient is derived consistently from the laminate A-matrix and closed-form width integrals for the assumed sinusoidal mode.

8.2 Amplitude equation and Den2

The post-buckling amplitude a(P) is computed from the difference (P - Pcr), scaled by an effective work-slope integral Den2. Den2 is computed directly from alpha0(x) and the squared slope of the longitudinal mode F'(x). A larger Den2 indicates stronger coupling between the applied load field and the local mode, leading to a larger post-buckling response for a given load increment.

8.3 Quartic coefficient k4

The stabilizing membrane stretching effect is represented by a quartic coefficient k4. k4 is assembled from face A-matrix terms (A11, A12, A22, A66) and exact width integrals for the sinusoidal phi(y). Using closed-form integrals avoids ad-hoc numerical factors and ensures consistent scaling when the panel width varies along the span.

8.4 Deflection decomposition used for plots

The load-deflection curve shown in FULL mode is decomposed into:

- Linear bending deflection: delta_lin = P * fb, where fb is the cantilever compliance based on EI_total(x).
- Local post-buckling increment: delta_loc = L * theta_fac * a(P), where theta_fac relates the mode amplitude to an equivalent tip deflection contribution.
- Total: delta_tot = delta_lin + delta_loc.

For consistent visualization, the default plot axes span 0 to 1.5 times the critical point (Pcr and delta_cr).

9. Numerical Implementation Details

9.1 Discretization

All spanwise integrals are evaluated on a uniform x-grid. Two parameters control the grid density:

- PPW: points per longitudinal wavelength (controls resolution relative to the expected mode wavelength).
- nx_min: minimum number of x grid points (absolute floor for robustness).

The solver estimates an initial wavelength from a width-based beta guess and selects:

nx = max(nx_min, ceil((L / lambda_guess) * PPW))

Typical defaults are PPW = 60 and nx_min = 1801 for FULL runs, and reduced values for UQ sweeps.

9.2 Integration scheme

All x-integrals are evaluated using the trapezoidal rule. This choice is stable, fast, and sufficiently accurate when PPW and nx_min are set to recommended values.

9.3 Mode normalization

Ritz modes have an arbitrary scale. For numerical stability and consistent post-processing, the solver normalizes the longitudinal mode based on the L2 norm of F'(x). This normalization makes Den2 and k4 comparisons less sensitive to eigenvector scaling and improves numerical robustness when exporting contours and deflection grids.

9.4 Stability guards

To prevent division by zero and improve robustness under extreme parameter variations, the implementation applies small positive floors (e.g., 1e-18) to denominators such as b(x), I_total(x), EI_total(x), and eigenproblem matrices. When k4 or Den2 becomes non-positive (due to inputs outside intended ranges), the post-buckling amplitude is suppressed to avoid numerical artifacts.

10. Solver Modes and Outputs

10.1 FULL (single-case)

FULL reads the Inputs sheet, runs the selected core (m2 or m3), and writes the Results sheet. For m3, it also generates plots and a DeflectionGrid export for downstream reporting or validation.

- Results sheet: Pcr, delta_cr, alpha*, beta*, lambda_x*, core, and Ktheta_root_per_m (for m3).
- Full_LoadDeflection.png: linear, nonlinear-only, and total load-deflection curves.
- Contour.png: normalized local deflection contour w(x, y) / max|w|.
- DeflectionGrid sheet: (x, y, w_norm) values on the requested grid resolution (e.g., 100x50).

10.2 SENS (OAT parameter sweep)

SENS performs one-at-a-time sweeps around the baseline Inputs values. Each selected parameter is perturbed over a user-defined range and evaluated independently. The solver core used for each evaluation is controlled by the 'core' setting (m2 or m3) to keep comparisons consistent.

SENS sheet schema (per row):

- name: input key to vary (must match a key in Inputs).
- mode: 'percent' (baseline +/- delta_percent) or 'range' (explicit low/high).
- delta_percent or low/high: perturbation definition.
- n_points: number of points in the sweep grid.
- enable: 1 to run, 0 to skip.

Outputs are written to M3_SENS_Results.xlsx, including a per-sample table and a baseline sheet.

10.3 SOBOL (Saltelli-lite UQ)

SOBOL computes variance-based global sensitivity indices using two base sample matrices A and B of size N_base x k, where k is the number of enabled uncertain inputs. For each variable i, an additional matrix AB_i is formed by replacing column i of A with column i of B. The total number of model evaluations is (2 + k) * N_base.

The solver reports first-order (S_i) and total-order (S_Ti) indices using standard Saltelli estimators and generates a bar chart sorted by S_Ti.

11. Verification and Example Cases

11.1 FEM benchmark verification

The solver has been benchmarked against a linear eigenvalue buckling analysis in ABAQUS for representative tapered box-beam geometries and composite layups. Verification focuses on two primary comparison metrics:

- Critical load agreement: relative error in Pcr between E3B and FEM.
- Mode similarity: qualitative agreement of the local face-buckling mode distribution (location of peaks and half-wave pattern).

In the provided benchmark example, the E3B prediction matched the FEM eigen-buckling load within approximately 2%, and the mode distribution was visually consistent.

11.2 Example case interpretation (REF / QI16 / QI8 / CROSS)

Example cases are intended to illustrate how laminate layup and stiffness distribution affect local buckling resistance and sensitivity rankings.

- REF: baseline layup used for comparison and parameter tuning.
- QI16 and QI8: quasi-isotropic-like layups with different balance of bending and membrane stiffness; compare changes in Pcr and in the post-buckling slope.
- CROSS: cross-ply dominant layup; typically shows lower shear coupling and can produce significantly different Pcr and mode localization.

When interpreting results, check (i) whether the mode remains root-localized, (ii) whether the load-deflection curve shows a clear nonlinear contribution after Pcr, and (iii) whether sensitivity/UQ rankings align with engineering intuition (e.g., thickness and width often dominate).

12. Limitations and Recommended Use

12.1 Modeling limitations (assumption-driven)

- Single half-wave width mode: higher-order width modes are not represented.
- Simplified edge restraint: webs provide rotational stiffness; flange restraint is neglected.
- Simplified post-buckling: Koiter-type amplitude provides a theoretical trend, not a full nonlinear equilibrium path.
- Section property mapping: thin-wall EI and alpha0 approximations are used instead of a full coupled beam-shell solution.
- No imperfection modeling: real structures may buckle below the theoretical Pcr due to imperfections.

12.2 Recommended defaults and parameter ranges

- FULL: PPW = 60, nx_min = 1801-2001 (good accuracy for plots and contours).
- SOBOL: PPW = 30, nx_min = 801 (faster evaluation for sampling runs).
- N_base: 200 as a practical default for quick ranking; increase for converged indices.
- Contour grid: 100x50 for reporting; increase to 120x60 or 160x80 for smoother contours.

12.3 When to prefer M2 vs M3

- Use M3 for final reporting, benchmark comparisons, and cases where root restraint (Ktheta_root_per_m) is important.
- Use M2 for rapid screening, early design iteration, and large parametric sweeps where speed matters more than boundary fidelity.
- For sensitivity/UQ, keep the same core across all cases to ensure comparable rankings.

Appendix A. Input Key Reference

The table below lists the primary input keys used by the solver. Keys are typically provided through the Excel Inputs sheet and, for UQ workflows, through the SENS and UQ sheets.

Key	Unit	Description	Typical range / example	Used in Section
case_name	-	Case identifier used in plot titles and file naming.	Any string	10
core	-	Solver core selection: m3 (default) or m2 (fast).	m2 or m3	10
L	m	Beam length (root to tip).	0.2 - 2.0	2,5,6
b_root	m	Interior panel width at root.	0.02 - 0.30	2,5,6
b_tip	m	Interior panel width at tip.	0.005 - b_root	2
h_root	m	Web height at root.	0.01 - 0.20	2,5,7
h_tip	m	Web height at tip.	0.005 - h_root	2,7
w_f	m	Flange width (each side).	0.0 - 0.05	2,5
Ef	Pa	Fiber Young's modulus.	50e9 - 350e9	3
Gf	Pa	Fiber shear modulus.	10e9 - 60e9	3
nuf	-	Fiber Poisson's ratio.	0.15 - 0.30	3
Em	Pa	Matrix Young's modulus.	1e9 - 6e9	3
num	-	Matrix Poisson's ratio.	0.25 - 0.45	3
Vf	-	Fiber volume fraction.	0.3 - 0.7	3
face_angles	deg	Face ply angles (bottom to top).	e.g., 0,45,-45,90,...	4
web_angles	deg	Web ply angles (bottom to top).	e.g., 0,45,-45,90,...	4
t_face_total	m	Total face laminate thickness.	0.0003 - 0.005	4,5,6
t_web_total	m	Total web laminate thickness.	0.0003 - 0.005	4,5,7
Ktheta_root_per_m	N·m/m	Root rotational spring per unit width (M3).	1e6 - 1e10	7
PPW	-	Points per wavelength for x	20 - 80	9

		discretization.		
nx_min	-	Minimum x grid size.	401 - 4001	9
grid	-	Contour grid resolution (nx x ny).	e.g., 100x50	10
N_base	-	Sobol base sample size (UQ_Control).	100 - 2000	10
seed	-	Random seed (UQ_Control).	Any integer	10

Appendix B. Output Fields Reference

Outputs depend on the selected mode. The most common files and fields are summarized below.

Output type	**Name / field**	**Meaning**	**Mode**
Results sheet	P_cr [N]	Critical local buckling load.	FULL
Results sheet	delta_cr [m]	Critical tip deflection at Pcr on the total curve.	FULL (m3)
Results sheet	alpha*, beta*, lambda_x*	Ritz parameters at the minimizing point and implied wavelength.	FULL (m3)
Image	Full_LoadDeflection.png	Load-deflection plot (linear, nonlinear-only, total).	FULL (m3)
Image	Contour.png	Normalized buckling mode contour.	FULL (m3)
Sheet	DeflectionGrid	Tabulated (x, y, w_norm) values for the contour grid.	FULL (m3)
Workbook	M3_SENS_Results.xlsx	Per-parameter sweep results and baseline Pcr.	SENS
Image	M3_Sobol_bars_core_<core>.png	Sobol bar chart of S_i and S_Ti sorted by S_Ti.	SOBOL
JSON (CLI stdout)	P_cr, delta_cr, plot paths	Summary object printed by the CLI driver.	All

Appendix C. Key Equations Summary

This appendix lists the key relations used by the solver in a compact form. See the main text for definitions and implementation details.

- Assumed mode: w(x, y) = F(x) * sin(pi y / b(x)), k_y(x) = pi / b(x).
- Eigenproblem (M3): N c = P * D c, with Pcr = min eigenvalue over scanned (alpha, beta).
- Edge-rotation stiffness (two webs): k_theta_web_pair(x) = 2 * (4 * D22_web / h(x)).
- Load mapping weight: alpha0(x) proportional to (L - x) and inverse of I_total(x).
- Koiter amplitude (for P > Pcr): a(P) = sqrt(((P - Pcr) * Den2) / k4).
- Den2: integral of alpha0(x) * (F'(x))^2 scaled by b(x)/2.
- Deflection: delta_lin = P * fb; delta_loc = L * theta_fac * a(P); delta_tot = delta_lin + delta_loc.