Supplementary methods and data

From distant volcanoes: obsidian and early Neolithic connections on the Şanlıurfa plateau

David J. Schwartzman1*, Robert Chis-Ciure1, Richard Hall2

1. Sussex Centre for Consciousness Science and School of Engineering and Informatics, University of Sussex, UK.

2. Hitchin Boys' School, Grammar School Walk, Hitchin, Hertfordshire, UK.

*Corresponding author: David J. Schwartzman, d.schwartzman@sussex.ac.uk

These materials document the evidence selection, checks on published data, model settings and sensitivity tests, and identify the files needed to reproduce the analysis.

S1 Evidence selection

The audit covers selected cases on the Şanlıurfa plateau and in the adjoining Taş Tepeler research landscape, identified through published site descriptions; Euphrates-valley cases were recorded separately. No inclusion polygon or fixed distance threshold is documented. The audit is not exhaustive.

A retained local case required a distinct locality, an explicit obsidian report and a reported Pre-Pottery Neolithic (PPN) component. Verified coordinates and geochemical source assignments were not required. A PPN attribution to a site does not securely date obsidian recovered from its surface. Broadly Neolithic survey material was not automatically assigned to the PPN, and shared modern village names were not sufficient to merge archaeological localities.

Akarçay Tepe and Mezraa-Teleilat were treated as Middle Euphrates valley cases outside the local plateau count despite published PPN obsidian. Nevalı Çori was also recorded separately as a valley case. Boncuklu Tarla lies outside the Şanlıurfa study area. The selection record gives the individual reasons for these decisions and for other exclusions or unresolved cases. It also distinguishes the three Yusuf Harabesi localities and records aliases including Besta Faki/Tahtari and Çakmak Harabesi/Harbetsuvan.

The search began with the indexed project corpus and bibliography, followed by bilingual web searches, citation chaining and checks of journal, author, conference and official project records. The query list was reconstructed from the search families used. It is not a complete contemporaneous search log: result counts, the exact order of every query and a full record of search-engine settings were not retained. We do not claim to have searched a bibliographic database or every source host exhaustively.

OpenAI Codex (GPT-5, GPT-5.6-sol and GPT-6) assisted candidate-source searches and compilation of the evidence register. The authors assessed the source claims; the manuscript's generative artificial intelligence statement describes the tool's wider use.

Documented source hosts include DergiPark; the Documenta Praehistorica publisher archive; ResearchGate and Academia.edu author deposits; official Taş Tepeler pages; Şanlıurfa municipal museum and governorship records; Turkish Ministry of Culture publications; SAA, ASOR and World Neolithic Congress records; Persée; ex oriente; and the University of Chicago archive. Linked sources are preserved in the selection record and the evidence and coordinate registers. Access to a source on one of these hosts does not establish an exhaustive search of that host.

The recorded query families, inclusion decisions, aliases and limits of individual source checks are provided in data/evidence/Supplementary_selection_record.pdf and data/evidence/selection_record.json. The 24 local cases comprise one published group-level comparison, two reported analyses without independently checkable artefact-level datasets and 21 presence cases without source attribution. These categories describe the available evidence, not the archaeological importance of the sites.

S2 Published data and transcription checks

Published source assignments and completeness of the published dataset are assessed separately. The "Published assignments" category requires an analytical publication that presents supporting measurements or geochemical comparisons; an incidental source mention is insufficient. Körtik Tepe and Hasankeyf Höyük provide itemised source assignments. The Jerf el Ahmar article reports source totals, analytical comparisons and plots, but not a complete register of its 44 artefacts. Table 1 uses the short label "Published assignments". Table S1's README worksheet explains the machine-readable category strings used by the code; those strings do not certify complete data availability.

The Körtik Tepe transcription contains 120 artefact records. Context fields, chemical entries and artefact descriptions were checked against the printed tables. Published signs and brackets are retained without assigning them an unsupported analytical meaning. The tables contain 68 Bingöl B assignments, whereas the narrative reports 67; the register retains the table's 68. Source assignments and the printed outliers remain unchanged. The detailed transcription record identifies the corrected cells.

Carter et al. (2021: Table 4) print a top-layer total of 27, although the counts for individual sources sum to 25: 19 Nemrut Dağ, five Bingöl B and one Group 3d. The Bingöl A and Muş cells are blank in the source and represented as zero in Table S1. Hasankeyf_Context preserves the reported total and gives the calculated sum separately, with a difference of two. The reported context subtotals sum to 334; the counts for individual sources sum to 332, matching the published grand total, including 307 Nemrut Dağ assignments. No pieces have been reassigned to reconcile the discrepancy.

Carter et al. (2021) also mention Bingöl and Nemrut Dağ obsidian at Göbekli Tepe, referring to MAX Lab work. The local evidence history and citation records include this mention, but the paper does not publish the underlying Göbekli artefact-level dataset. Göbekli Tepe is classified as reported analysis.

The Harbetsuvan source checks cover the author-deposited thesis (Şen 2022) and the full article (Şen and Çelik 2022), including their tables and graphical material. They distinguish the wider assemblage of 1778 finds from the 177 production-debris pieces selected for chemical analysis. The checked files provide no individual chemical source assignments. Details, file identifiers and discrepancies in published summary values are recorded in data/evidence/Harbetsuvan_source_verification.pdf; comparator checks are recorded in data/evidence/Comparator_transcription_verification.pdf.

S3 Scenario matrix

The matrix contains 18 local paths, comparing the same three source alternatives at each of six destinations, and four comparator paths. Harbetsuvan, Karahan and Göbekli are the principal destinations; Sefer, Taşlı and Hamzan are supplementary. Sefer's locality coordinates are verified; Taşlı and Hamzan retain illustrative endpoints. Other possible sources were not tested.

Figure 2 displays seven principal paths, three supplementary paths and four comparator paths. The two principal paths not drawn are Nemrut Dağ–Karahan and Nemrut Dağ–Göbekli. Table S2 reports all 22 calculated paths.

S4 Model settings and distance measurement

The calculations retain the original model's endpoints, point order, terrain tiles, grids and movement costs. The base model uses the zoom-8 Web Mercator grid, a downsampling factor of 4 with non-overlapping arithmetic block means, and a 0.9° search margin. Median effective cell dimensions are 1.93 × 1.93 km. Distances between cell centres come from the original row-distance tables. Dijkstra's algorithm searches a 5 × 5 movement neighbourhood. Moves spanning several cells are expanded into adjacent steps by the original Bresenham function, so intervening cells contribute to the cost.

All 22 base searches and 54 sensitivity searches run from source to site; no reverse searches were performed. The original run_route function and the ordered query pairs in complete_source_matrix.py specify this direction. The C++ helper reads those same ordered pairs.

Movement cost starts with the direction-dependent Tobler speed function, $6 \times \exp[-3.5 \times |\text{slope} + 0.05|]$, with a minimum speed of 0.35 km/h. The model then applies the original adjustments for ruggedness, elevation above 2,000 m, topographic position, ridges, concentrated drainage and valley floors. The exact formulas and thresholds are in the original model script. After these adjustments, accumulated cost is a relative composite index, not journey duration.

Coordinates and permitted moves

The raster uses the spherical Web Mercator tile scheme. Source and site coordinates are latitude/longitude pairs in decimal degrees; the script obtains cell-centre coordinates through

its inverse tile transformation. Distances are calculated from these geographic coordinates, not from projected Mercator metres. The original build_distance_tables function uses the haversine formula and an Earth radius of 6,371 km to calculate east–west spacing along each row and north–south spacing between successive rows. These spacings are stored as float32 values. Horizontal and vertical steps use the stored spacings directly. A diagonal step uses the hypotenuse calculated from the east–west and north–south spacings at the lesser row index. Diagonal distances are therefore local approximations, rather than separate haversine calculations between diagonal endpoints. Endpoint connectors use direct haversine distances between their geographic coordinates.

The base search permits 24 offsets: row and column differences from −2 to +2, excluding (0, 0), within the grid boundaries and where edge costs are finite. The bresenham_cells function divides longer moves into adjacent-cell steps and sums their direction-dependent costs. No intervening cells are bypassed. The 3 × 3 sensitivity setting permits only the eight adjacent-cell offsets. A candidate route replaces the current one only if its cost is strictly lower, and the search stops when the goal is removed from the priority queue. Because the calculation preserves the sign of the slope, reversing the endpoints can change the cost. Reverse paths were not calculated.

How the terrain terms combine

For an elementary directed step from cell i to cell j, let d be its horizontal distance in kilometres and $s = (\text{elevation}_j - \text{elevation}_i) / \max(1{,}000d, 1)$, with elevation in metres. Walking speed is $v = \max(0.35, 6 \exp[-3.5 |s + 0.05|])$ km/h. In the base model, the step cost is $(d/v) \times A \times R \times L \times C \times V$. These step costs are summed along the path. The factors are defined below. Each average is the arithmetic mean of the values at the two cells.

- Altitude and steepness (A): start at 1; above an average elevation of 2,000 m, add $\min((\text{elevation} - 2{,}000)/2{,}200, 0.85)$. Multiply this factor by 2 where $|s| > 0.55$. An average elevation below −10 m gives infinite cost and excludes the step.
- Ruggedness (R): $1 + \min(\text{relief}/420, 1.35)$. Relief at each cell is the elevation range in its 3 × 3 neighbourhood.
- Landform (L): topographic position is cell elevation minus the mean elevation in a 9 × 9 neighbourhood. If its two-cell average t is below −25 m, $L = \max(0.84, 1 + t/650)$. If t exceeds 45 m, $L = 1 + \min(t/700, 0.35)$; otherwise L = 1.
- Concentrated drainage (C): let $f = \ln(1 + \text{average flow accumulation})$. If $f > 5.5$, $C = 1 + \min((f - 5.5) \times 0.18, 0.7)$; otherwise C = 1.
- Valley preference (V): if $t \le 5$ m, $V = 1 - \min(\max(f - 1.5, 0) \times 0.04, 0.12)$; otherwise V = 1.

The weaker and stronger terrain variants replace each factor M separately with $1 + (M - 1)w$, where w = 0.75 or 1.25; the base value is w = 1. The factors are still multiplied together. These variants leave the Tobler term and elevation exclusion unchanged, and the resulting cost remains a composite index rather than elapsed time.

Drainage and river crossings

Flow accumulation is calculated from the elevation grid, independently of the river lines shown on the map. Each cell starts with a contribution of one and passes its accumulated contribution to the steepest strictly downhill neighbour among the eight adjacent cells. If there is no lower neighbour, the cell retains its accumulation. The code neither fills depressions nor routes flow across flat areas. The resulting count indicates drainage concentration in the terrain; it does not measure water discharge or river width.

High-flow cells receive the finite C penalty, up to 1.7 in the base model. Low topographic position can also activate the valley preference. The penalty applies whenever a path passes through those cells, not specifically when it crosses a mapped channel. Natural Earth and OpenStreetMap/Overpass waterways are shown for context and do not enter the cost calculation. The model has no river or lake mask, channel-width or depth threshold, or explicit rule for fords, bridges or crossing points. Excluding steps with average elevation below −10 m does not amount to a water mask. A path crossing a mapped waterway therefore does not establish a feasible crossing. Boat travel and combined land-and-river journeys are not modelled.

The original implementation is documented by deg2num, pixel_to_lat, pixel_to_lon, build_distance_tables, compute_ruggedness, compute_topographic_position, compute_flow_accumulation, bresenham_cells, edge_cost, dijkstra_path and run_route, together with the complete-matrix query setup and C++ search.

Full-path distance and endpoint connectors

Reported path length sums the horizontal distances of all elementary costed grid steps, including intervening cells in longer moves, and the two endpoint connectors. These connectors are haversine distances from the exact source and site coordinates to their grid-cell centres. Both are included in path length, even when shorter than 0.01 km, but neither contributes to cost or route optimisation. Lines are simplified only for display. The metric definitions distinguish this measure from diagnostic parent-chain and display-line lengths. Percentages use the unrounded analytical totals. Table S2 expresses cost relative to the base Göllü Dağ/Kömürcü–Harbetsuvan value of 117.82295375584469, set to 100.

S5 Sensitivity tests and coordinate checks

Each of the six sensitivity variants was applied to all nine principal paths, giving 54 runs. Each variant changes only the setting listed below; all other settings remain at their base values.

- Higher resolution: Downsampling factor 3 instead of 4.
- Movement representation: 3 × 3 instead of 5 × 5 neighbourhood.
- Narrower search extent: Margin 0.45° instead of 0.9°.
- Wider search extent: Margin 1.35° instead of 0.9°.
- Weaker terrain adjustments: Non-Tobler multiplier 0.75 instead of 1.0.

- Stronger terrain adjustments: Non-Tobler multiplier 1.25 instead of 1.0.

Path lengths change by −3.08% to +4.72%. Both eastern source alternatives remain shorter than Göllü Dağ/Kömürcü at all three principal destinations. The 3 × 3 and 5 × 5 settings give identical expanded paths and lengths for all nine pairs. This checks movement representation rather than geographical robustness. All nine paths are also unchanged under the two alternative search margins. These sensitivity tests cover only the nine principal paths, not the supplementary local or regional comparator paths.

The sensitivity tests do not assess source-endpoint placement or the exact locations of Taşlı and Hamzan. The model uses present topography and drainage. It does not reconstruct Early Holocene vegetation, palaeohydrology, seasonality, snow or soils, or include boat travel or social constraints. A shorter path or lower composite cost cannot identify the source used, the route travelled or the way material was transferred.

Coordinate cross-check

The independently verified locations of Harbetsuvan, Karahan, Göbekli and Sefer fall within their original base-grid cells. Replacing only their terminal connectors changes the total lengths of the 12 paths to these sites by less than 0.06%. The six principal percentage contrasts change by less than 0.02 percentage points. For Karahan–Nemrut, a change below 0.01 percentage points crosses a rounding boundary, changing the displayed value from 30.8% to 30.7%. Reported principal values continue to use the original endpoints. See the coordinate-check results.

S6 Supplementary files and access

The supplementary files are supplied in Supplementary_material.zip. Verify its checksum and use the files together when reproducing the analysis. The interactive map at https://obsidianmap.netlify.app/ presents an earlier package; use the accompanying archive for the analysis reported in the paper.

The interactive map is hosted on Netlify. The supplementary package is available from Zenodo at https://doi.org/10.5281/zenodo.23318701. The files also accompany the submission for review.

All file paths below start at the top level of the extracted ZIP, which contains the data/, source/ and figures/ directories. package_metadata.json describes the package components, access and repository status. build_manifest.json gives SHA-256 checksums for the files; FILE_INVENTORY.tsv lists their paths, sizes and checksums. The ZIP's own checksum is supplied separately as Supplementary_material.zip.sha256. SUPPLEMENTARY_FILE_GUIDE.pdf contains the supplementary methods and file guide reproduced here.

The package contains the complete source–site matrix, sensitivity results and the model code and terrain data needed to reproduce them.

Archaeological evidence and selection records

- Table S1: data/Supplementary_Table_S1.xlsx. Local_Cases contains the 24 retained local cases, and Comparator_Coordinates gives the locations of the four regional comparators. The other worksheets are README, Evidence_Register, Harbetsuvan_XRF, Kortik_Artefacts, Hasankeyf_Context, Hasankeyf_Tools, Source_Register, Site_Context_Core, Citation_Status and Research_Gaps. Rows represent different units—individual artefacts, analytical groups, contexts and bibliographic notes. Count localities from Local_Cases, not from the total rows across worksheets.
- Map evidence records: data/evidence/local_cases.json and source/local_cases.json contain the 24 local records exported from Table S1's Local_Cases worksheet. Run source/verify_complete_matrix.py to compare both JSON copies with the worksheet.
- Coordinates: data/evidence/coordinate_audit.json and data/evidence/coordinate_audit.csv document the checks on local site locations. source/map_points.json contains the map points. Verified site locations are recorded separately from the endpoints used to calculate the paths.
- Selection and search records: data/evidence/Supplementary_selection_record.pdf explains the selection process. data/evidence/selection_record.json contains the recorded searches and candidate decisions in selection_record, and the selection criteria and targeted source check in assessment_addendum. The search terms were reconstructed from retained records; they are not a complete log made at the time of the searches.

Path results and full route coordinates

- Table S2: data/Supplementary_Table_S2.xlsx gives all 22 base scenarios. source/map_table_s2.json provides the definitions, unrounded values, analytical roles and six principal source contrasts used by the map.
- Inputs: data/complete_matrix/complete_matrix_inputs.json lists the 22 source–site pairs, their coordinates and the seven model configurations. The base analysis has nine principal local paths, nine supplementary local paths and four regional comparator paths. The local comparison covers all three selected source zones, but does not test every possible source.
- Full base paths: data/complete_matrix/primary_complete_paths.json contains all 22 base paths. Each record includes parent_grid_cells, expanded_grid_cells and expanded_grid_coordinates, the exact source and site coordinates, and the coordinates of the simplified display line.
- Distances and costs: data/complete_matrix/distance_metrics_all_76.csv and data/complete_matrix/distance_metrics_all_76.json contain the 22 base calculations and 54 principal sensitivity calculations. For each path, grid_path_length_km + source_connector_km + target_connector_km = total_path_length_km. Grid length includes every elementary costed step, including those within longer moves. The

connectors add distance but do not contribute to reproduced_composite_cost. data/complete_matrix/distance_metric_definitions.json distinguishes these analytical measures from parent-chain, haversine-diagnostic and simplified-display lengths.

- Principal comparisons: data/complete_matrix/source_contrasts.csv and data/complete_matrix/source_contrasts.json give six within-site contrasts for each of the seven configurations, making 42 comparisons. The nine principal paths connect the selected source endpoints to Harbetsuvan, Karahan and Göbekli. Paths to Sefer, Taşlı and Hamzan are supplementary. The Taşlı and Hamzan endpoints are illustrative and are excluded from the principal analysis.
- Coordinate check: data/complete_matrix/verified_coordinate_check.csv and data/complete_matrix/verified_coordinate_check.json compare terminal connectors at the four local model destinations whose coordinates were independently verified. The reported results use the original endpoints.

Sensitivity results and path reconstruction

The six sensitivity files are in data/complete_matrix/. Each contains the full paths for all nine principal pairs, as well as the simplified display lines:

- higher_resolution_complete_paths.json: downsampling factor 3.
- three_by_three_neighbourhood_complete_paths.json: eight adjacent-cell moves.
- narrower_search_margin_complete_paths.json: 0.45° margin.
- wider_search_margin_complete_paths.json: 1.35° margin.
- weaker_non_tobler_terms_complete_paths.json: terrain-modifier strength 0.75.
- stronger_non_tobler_terms_complete_paths.json: terrain-modifier strength 1.25.

The same directory contains sensitivity_length_changes.csv and sensitivity_length_changes.json, which compare all 54 sensitivity runs with their base results. analysis_summary.json records the reproduction checks.

Calculation code, interactive map and figures

- Calculation code: source/complete_matrix/complete_source_matrix.py and source/complete_matrix/exact_dijkstra.cpp calculate the paths. source/complete_matrix/check_verified_coordinates.py checks the coordinates. Commands and requirements are given in Levant_Supplementary_Methods.pdf. The calculation requires Python, NumPy, Pillow and a C++17 compiler named clang++.
- Terrain model: source/terrain_model/ contains generate_least_cost_paths.py, least_cost_path_inputs.json, least_cost_path_outputs.json, least_cost_path_sensitivity.json and requirements.txt. Its terrain_cache/ directory contains the cached Mapzen Terrarium tiles. The complete-matrix calculation uses this model code and cached terrain tiles; the original output and sensitivity files support checks against the model's earlier calculations. The original simplified lines record the model's display geometry; reported analytical distances use the full paths.

- Interactive map: index.html is built by source/build_map.py from source/map_template.html, source/local_cases.json, source/map_points.json, source/map_routes.json and source/map_sourceIndex.json. The map reports all 22 calculations; each route's metadata identifies its analytical role and whether it appears in Figure 2. Leaflet and the basemaps require an internet connection, whereas terrain calculations use the supplied cached tiles.
- Figure 1: source/figure1_reproduction/source/create_figure_1_locator_map.py, with instructions in source/figure1_reproduction/README.pdf.
- Figure 2: source/figure2_reproduction/source/create_figure_2_presentation.py, with instructions in source/figure2_reproduction/README.pdf. Run python source/figure2_reproduction/reproduce_figure_2.py from the extracted package to regenerate the figure. It uses the original 14-path display output at source/figure2_reproduction/data/least_cost_path_outputs.json. figures/figure_manifest.json lists the displayed paths, which are a subset of the 22 calculated paths.
- Figure requirements: each reproduction directory includes source/requirements_figures.txt, source/coordinate_audit.json and source/figure_data_cache/. The Figure 2 directory also contains the Figure 1 plotting helper. The PDF guides identify the font needed to reproduce the typography exactly.
- Figure files and captions: figures/Figure_1.png and figures/Figure_2.png are also supplied as .pdf and .svg files. Captions are in figures/figure_captions.json. figures/figure_manifest.json gives the file checksums and lists Figure 2's 14 scenarios.

Validation and supporting files

From the extracted package, run python source/verify_complete_matrix.py to check the 22 paths, 54 sensitivity runs, six principal contrasts, map values, Figure 2 subset and file checksums. The command also checks agreement between Table S1 and the JSON records. This command checks the supplied outputs; it does not recalculate terrain paths.

Table S2 and data/complete_matrix/ contain the complete calculated matrix. The package excludes private manuscripts, credentials and full-text source-paper PDFs.

S7 Reproducing the calculations

The six *_complete_paths.json files stored beside the base output contain all principal sensitivity paths. Historical values are present only for the 56 scenarios in the original model outputs retained in source/terrain_model/. A missing historical value means the scenario had not been calculated; it must not be read as zero. The field archived_display_coordinates holds display coordinates for all scenarios; historical_scenario distinguishes those in the original model from those added to complete the matrix.

The calculation script uses the unchanged scientific model to build the grids and calculate edge costs. Its C++17 Dijkstra helper preserves the original neighbour and heap order, strict relaxation, closed-set handling and binary64 addition, without fast-math. It reproduces every

original cost and display geometry. Each path is checked by summing its elementary movement costs. Checksums taken before and after calculation confirm that the model files are not modified by the calculation.

To run the analysis, use Python with NumPy and Pillow, a C++17 compiler named clang++, and the original cached terrain tiles supplied in the package. Run these commands from the package's top-level directory:

python source/complete_matrix/complete_source_matrix.py --original source/terrain_model --out calculation_results

python source/complete_matrix/check_verified_coordinates.py --site . --analysis calculation_results

python source/verify_complete_matrix.py

The calculation uses only the cached terrain tiles and makes no network requests. Local calculation caches speed up the run but are not included in the published package. Table S2 and data/complete_matrix/ contain the results reported in the paper.

References for the data checks

Abbès, F., Bellot-Gurlet, L., Cauvin, M.-C., Delerue, S., Dubernet, S., Poupeau, G. and Stordeur, D. 2003. Provenance of the Jerf el Ahmar (Middle Euphrates Valley, Syria) obsidians. Journal of Non-Crystalline Solids 323: 162–66. https://doi.org/10.1016/S0022-3093(03)00299-0

Carter, T., Grant, S., Kartal, M., Coşkun, A. and Özkaya, V. 2013. Networks and Neolithisation: sourcing obsidian from Körtik Tepe (SE Anatolia). Journal of Archaeological Science 40 (1): 556–69. https://doi.org/10.1016/j.jas.2012.08.003

Carter, T., Moir, R., Wong, T., Campeau, K., Miyake, Y. and Maeda, O. 2021. Hunter-fisher-gatherer river transportation: insights from sourcing the obsidian of Hasankeyf Höyük, a Pre-Pottery Neolithic A village on the Upper Tigris (SE Turkey). Quaternary International 574: 27–42. https://doi.org/10.1016/j.quaint.2020.09.045

Şen, Z. 2022. Harbetsuvan Tepesi obsidiyen aletlerin tipolojik ve teknolojik analizleri. MA thesis, Iğdır University. Author deposit.

Şen, Z. and Çelik, B. 2022. Çanak Çömleksiz Neolitik Dönem'e ait Harbetsuvan Tepesi obsidyen buluntuları. Arkeoloji ve Sanat 170: 13–22. Author record.