Mike's RF Line of Sight Tool — Technical Stack Overview

Developed by N4MXB

Architecture

The tool is a **single-file, zero-dependency web application** — one HTML file containing all HTML, CSS, and JavaScript inline. There is no build process, no npm, no bundler, no framework, and no backend server. It runs entirely in the browser. The only external dependencies are two CDN-loaded libraries (Leaflet and its CSS), two Google Fonts, and a small Cloudflare Worker that acts as a CORS proxy for third-party APIs that don't support browser-direct access.

This architecture was a deliberate choice: the tool can be hosted on any static file server, GitHub Pages, or run straight from a local filesystem — no server-side runtime required.

Mapping

Library: Leaflet.js v1.9.4 — the industry-standard open-source slippy map library for web applications.

Tile sources (user-selectable):

Layer	Source	Notes
Esri Satellite	server.arcgisonline.com — World Imagery MapServer	Default on load. Free, no API key.
Esri Hybrid	Same satellite layer + Reference/World_Boundaries_and_Places label overlay, composited as a L.layerGroup	Free, no key.
Google Satellite	mt1.google.com/vt/lyrs=s	Requires user-supplied Google Maps API key.
Dark	OpenStreetMap-based dark tile server	Free, no key.

Both markers (Site A and Site B) are custom SVG L.divIcon elements with colored glow effects, fully draggable. A "ghost marker" follows the cursor during map-click placement mode. The path between sites is rendered as a segmented L.polyline — green segments where LOS is clear, orange where terrain blocks the path.

Geocoding (Address → Coordinates)

Address lookup uses a **three-source cascade** with automatic fallback. Sources are tried in order; the first successful result with sufficient confidence wins.

1. **ArcGIS World Geocoder** (geocode.arcgis.com) — Primary. Best US street-level accuracy, free with no API key required. Results with a confidence score below 70 are discarded and the next source is tried.
2. **Photon by Komoot** (photon.komoot.io) — Secondary. OpenStreetMap-based, international coverage, CORS-safe.
3. **Nominatim / OSM** (nominatim.openstreetmap.org) — Last resort. The reference OSM geocoder; slower and more prone to ambiguous results but globally comprehensive.

Geocoding is also "lazy" — address fields are not resolved until the Calculate button is pressed, at which point geocodeIfPending() silently resolves any unconfirmed address before the elevation fetch begins.

Elevation Data

Elevation profiles are fetched as 100 evenly-spaced sample points along the great-circle path. 100 points is the maximum batch size for both APIs and provides approximately one sample every 0.35 miles on a 35-mile path.

Primary: [Open-Elevation]() — POST-based JSON API using SRTM (Shuttle Radar Topography Mission) data. Accepts a JSON body of {locations: [{latitude, longitude}, ...]}. Returns an array of elevation values in meters.

Fallback: [Open-Meteo Elevation API]() — GET-based, comma-separated lat/lon pairs. Also SRTM-derived. Used if Open-Elevation is unavailable or returns an unexpected response.

If both sources fail, the user receives a specific error message naming both services and suggesting they wait and retry.

Caching: Elevation results are cached in memory keyed by the endpoint coordinates rounded to 4 decimal places (~11 meter precision). Subsequent calculations with the same two sites (e.g. changing frequency band, adjusting antenna height, tweaking link budget) reuse the cached profile with zero API calls. The cache holds up to 5 unique paths (LRU-style eviction) to prevent unbounded memory growth.

Callsign & Repeater Lookups

All third-party callsign and repeater APIs require a server-side proxy to work around CORS restrictions. This is handled by a **Cloudflare Worker** (rf-los.myz06vette.workers.dev) — a small JavaScript function running at Cloudflare's edge network, globally deployed, free tier. The Worker routes requests based on a type= URL parameter.

Repeater Lookup

Source: RepeaterBook API

Queries by callsign with country=United%20States. For GMRS repeaters, appends &service=gmrs. Returns JSON with repeater details including frequency, offset, PL/CTCSS tone, location coordinates, mode (FM/DMR/etc.), and operational status. When a callsign has multiple associated repeaters, the tool presents a picker so the user can select the intended one.

Ham Licensee Lookup

Source: callook.info — Free REST API (/{CALLSIGN}/json) maintained by Josh Dick, W1JDD. Updated daily from FCC database snapshots. Notably returns **pre-geocoded latitude/longitude** (computed via Bing Maps from the mailing address), so no additional geocoding step is needed for ham licensees. Also returns the Maidenhead grid square.

GMRS Licensee Lookup

GMRS callsigns (format: W[A-Z]{3}[0-9]{3} for modern, KA[A-Z][0-9]{4} for legacy) are detected by regex and handled differently. **There is no public API for GMRS callsign lookup.** The FCC's data.fcc.gov License View API searches by company name only (and is a deprecated demonstration system). The real ULS at wireless2.fcc.gov blocks all non-browser requests with HTTP 403. callook.info and hamdb.org are amateur (Part 97) only and do not index GMRS (Part 95 / service code ZA) licenses.

For GMRS callsigns, the tool returns a structured informational response with a direct pre-filled link to the FCC ULS search for that specific callsign and service code — the most useful thing that can be done given the constraint.

Maidenhead Grid Square Lookup

No external API. Grid square decoding is implemented entirely in JavaScript from the ARRL/ITU Maidenhead Locator System specification. The algorithm decodes both 4-character (field + square) and 6-character (field + square + subsquare) locators:

- **Field** (chars 1–2, A–R): 20° longitude × 10° latitude blocks
- **Square** (chars 3–4, 0–9): 2° longitude × 1° latitude subdivisions
- **Subsquare** (chars 5–6, A–X): 5' longitude × 2.5' latitude subdivisions

The decoded center coordinate is placed on the map immediately. Precision notes are calculated and displayed: a 4-character grid resolves to approximately ±70 miles; a 6-character subsquare resolves to approximately ±3–15 miles.

RF Propagation Calculations

All calculations run in the browser in pure JavaScript. No libraries are used.

Constants

- **Earth radius:** 6,371,000 meters
- **Effective Earth radius factor:** k = 4/3 (the standard RF engineering approximation accounting for average tropospheric refraction, which bends signals slightly downward and effectively increases the radio horizon relative to geometric LOS)

Distance

Haversine formula — great-circle distance between two lat/lng pairs on a sphere of radius 6,371 km. Used for path distance display, intermediate point interpolation, and all diffraction calculations.

Earth Bulge

The midpoint of a long path sits physically higher than a straight line drawn between the two endpoints due to Earth's curvature. The bulge at distance d along a path of total length D is:

bulge(d, D) = d × (D - d) / (2 × k × R)

where k = 4/3 and R = 6,371,000 m. This bulge is added to terrain elevation at each sample point before comparing against the LOS line, so the tool correctly accounts for Earth curvature on long paths.

Frequency Bands

The tool uses representative center frequencies for each band selection. These drive wavelength (λ) for all diffraction calculations:

Band	Range	Center Freq Used	λ
VHF Low	30–88 MHz	60 MHz	5.00 m
VHF High	136–174 MHz	155 MHz	1.94 m
UHF Low	400–512 MHz	450 MHz	0.67 m
UHF High	700–900 MHz	800 MHz	0.375 m

Lower frequency = longer wavelength = more diffraction around obstacles. This is why a POOR result at VHF might still yield a marginal contact while the same path at UHF would be hopeless.

Fresnel-Kirchhoff Diffraction Parameter (v)

For each of the 100 terrain sample points, v is calculated as:

$v = h\_obs \times \sqrt{(2(d_1 + d_2) / (\lambda \times d_1 \times d_2))}$

where h_obs is the height of the obstacle above the LOS line (positive = above, negative = below), d_1 is the distance from Site A to the obstacle, d_2 is from the obstacle to Site B, and λ is wavelength. The worst (highest) v value across all 100 points is used for grading.

Single Knife-Edge Diffraction Loss — ITU-R P.526

The primary single-obstacle diffraction loss J(v) per the ITU-R P.526-15 piecewise approximation:

$v \leq -0.78 \rightarrow J = 0$ dB (full Fresnel clearance)

$v \leq 0 \rightarrow J = 20 \cdot \log_{10}(0.5 - 0.62 \cdot v)$

$v \leq 1 \rightarrow J = 20 \cdot \log_{10}(0.5 \cdot e\hat{}(-0.95 \cdot v))$

$v \leq 2.4 \rightarrow J = 20 \cdot \log_{10}(0.4 - \sqrt{(0.1184 - (0.38 - 0.1 \cdot v)^2)})$

$v > 2.4 \rightarrow J = 20 \cdot \log_{10}(0.225 / v)$

Note: J(v) returns a negative value by convention (loss is positive). The tool negates and clamps to zero when needed.

Link Quality Grading

The v value maps to five human-readable grades:

Grade	Condition	Description
EXCELLENT	LOS clear, $v \le -0.78$	First Fresnel zone fully clear
GOOD	LOS clear, $v \le 0$	Minor Fresnel intrusion
GOOD	LOS clear, $v \le 0.5$	Partial Fresnel zone blocked
FAIR	Any, $v \le 0.5$	Marginal LOS, reduced signal
FAIR	Any, $v \le 1.5$	Diffraction path, likely usable
POOR	Any, $v \le 2.5$	Heavy diffraction, marginal/intermittent
UNLIKELY	Any, $v > 2.5$	Severe obstruction, contact improbable

The "60% Fresnel clearance rule" ($v = -0.78$) is the classical criterion for treating a path as effectively clear — at that point, 60% of the first Fresnel zone radius is unobstructed, and diffraction loss is negligible.

Deygout 3-Obstacle Diffraction — ITU-R P.526-15 §4.2

The single knife-edge model systematically underestimates loss on Appalachian-style terrain with multiple successive ridges. When $v > 0.5$, the tool supplements with the Deygout method:

1. Find the **dominant obstacle** across the full path (highest v) — the _findPeak() function.
2. Find the dominant **left sub-obstacle** on the sub-path from Site A to the main peak.
3. Find the dominant **right sub-obstacle** on the sub-path from the main peak to Site B.
4. Sum all three individual knife-edge losses.

Each sub-path builds its own LOS reference line between its endpoints (not the original site-to-site LOS), correctly modeling the cascaded diffraction geometry. The extra loss beyond the single knife-edge result is what gets added to the corrected link budget.

Δh Terrain Roughness Penalty — ITU-R P.1546 / Longley-Rice

Even Deygout misses the distributed, background scatter loss from rough terrain — the rolling hills, rocks, and trees between the main peaks. The Δh interdecile height variation captures this:

$\Delta h = P_{90}(\text{elevation}) - P_{10}(\text{elevation})$ across all 100 sample points

The loss model, calibrated against Longley-Rice empirical outputs for VHF/UHF terrestrial paths:

$L\_\Delta h = 5.0 \times \log_{10}(\Delta h / 10) \times (1 + 0.3 \times \log_{10}(d\_km / 10)) \times (f / 150)^{0.1}$

Capped at 14 dB to prevent double-counting with Deygout corrections. Typical values:

- $\Delta h < 10$ m → 0 dB (flat terrain, negligible)
- $\Delta h \approx 50$ m → ~3 dB (rolling hills)
- $\Delta h \approx 150$ m → ~7 dB (significant terrain)
- $\Delta h \approx 300$ m → ~11 dB (mountainous)

The Deygout extra loss and Δh penalty are summed and capped at a combined maximum of **28 dB** to prevent physically unrealistic results on extreme terrain, then subtracted from the knife-edge margin to produce the **terrain-corrected margin**.

Link Budget

Transmitter Side

- **TX Output** — user input in watts
- **Feedline Loss** — entered as dB or as watts dissipated (converted internally: $dB = -10\cdot\log_{10}(1 - W\_lost / P\_tx)$)
- **TX Antenna Gain** — entered as dBd or dBi with live toggle; internally stored as dBd. Conversion: dBi = dBd + 2.15
- **ERP (Effective Radiated Power):** ERP_W = P_tx × 10^(dBd/10) × 10^(−feedline_loss/10) — the industry-standard ham radio power figure, referenced to a half-wave dipole
- **EIRP (dBm):** EIRP = P_tx_dBm + dBi − feedline_loss_dB — used internally for FSPL math (isotropic reference required)

Path Loss

Free Space Path Loss (FSPL):

$\text{FSPL (dB)} = 20\cdot\log_{10}(d) + 20\cdot\log_{10}(f) - 147.55$

where d is in meters and f is in Hz. The −147.55 constant consolidates the speed-of-light and unit conversion terms.

Received Signal Level

RSL (dBm) = EIRP (dBm) − FSPL (dB) − terrain_loss (dB)

Receiver Sensitivity Presets

Preset	Sensitivity
HT (handheld)	−107 dBm
Mobile radio	−110 dBm
Repeater input	−116 dBm
Sensitive receiver	−120 dBm
SDR	−130 dBm
Custom	User-defined

Link Margin & Verdict

Margin (dB) = RSL − RX Sensitivity

When terrain corrections apply (Deygout + Δh), these are subtracted to yield the **terrain-corrected margin**, which is the primary displayed figure. The original knife-edge margin is preserved as a secondary reference row in the budget table.

Verdict thresholds:

- ≥ 20 dB → Solid link, excellent margin
- ≥ 10 dB → Good link, adequate margin
- ≥ 0 dB → Marginal, expect static
- < 0 dB → Link budget fails
- Severe terrain block (v > 3.0, margin < 20 dB) → Terrain blocks this path (no power recommendation issued, as the geometry itself is the problem)

Elevation Profile Chart

The elevation profile is rendered on an **HTML5 Canvas 2D** element — no charting library. The chart is drawn from scratch on every analysis and on every resize event. Key features:

- Terrain fill with blocked segments rendered in orange, clear segments in green
- LOS line with antenna heights at each end
- Earth bulge represented as an upward offset on the terrain profile (visually bakes the curvature correction in)
- Interactive crosshair: mousemove events on the canvas track the cursor position, interpolate the corresponding map coordinate, and place a live marker on the Leaflet map at that exact point
- On mobile, the chart is rendered in a dedicated full-screen tab and redrawn at correct dimensions when that tab becomes active (avoiding the common canvas sizing bug where a canvas renders at zero size because it was hidden during the initial draw)

Typography & Styling

All styling is hand-written CSS with CSS custom properties (variables) for the color theme. No CSS framework.

- **Barlow** (Google Fonts) — UI labels, section headers, buttons. A condensed geometric sans-serif with high legibility at small sizes.
- **Share Tech Mono** (Google Fonts) — All numerical readouts, coordinates, stat values, callsign fields, and data-heavy elements. A monospaced typeface with a technical/radio operator aesthetic.

Primary accent color: #3DFFA0 (bright green). Warning/obstruction color: #ff6b35 (orange). Marginal color: #ffe066 (amber).

Performance & Reliability

- **Debounce:** runAnalysis() is debounced to 600ms. All input changes, marker drags, geocode callbacks, and repeater lookups funnel through a single timer, ensuring at most one elevation API call fires per user action regardless of how many things trigger simultaneously.

- **Concurrency guard:** _analysisRunning flag prevents re-entrant analysis if a previous fetch is still in flight.
- **API retry logic:** The Open-Meteo fallback retries twice with a 1.5-second delay on 429 responses before giving up and letting the error propagate to the error handler.
- **Responsive layout:** Below 768px the UI switches to a 4-tab mobile layout (Map / Profile / Controls / Results). The desktop layout is a fixed left-panel + full-height map split. No media-query breakpoint hacks on the canvas — chart dimensions are measured from the live DOM at draw time.