Assignment 3
ELG3106 – Fall 2025

Section: A00
Name: Robert Gauvreau

Date Assigned: Sunday October 20th, 2025
Due Date: Saturday November 5th, 2025

Table of Contents

Introduction:

A microstrip line is a type of transmission line formed by a conducting strip of width w on a dielectric substrate of thickness h with relative permittivity ε_r, which rests on a ground plane. The electromagnetic field extends partly in the dielectric and partly in air as displayed in *Figure 1.*, the supported mode is quasi-TEM and is typically modeled with an effective relative permittivity ε_{eff}. Using ε_{eff}, wave propagation along the line can be approximated by the TEM relations, in particular, the phase speed is

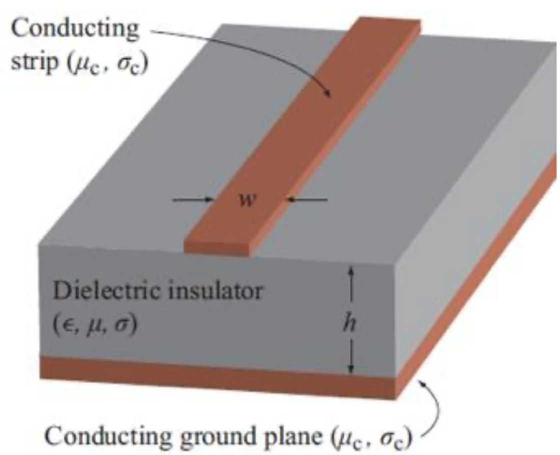

Figure 1 Microstrip line geometry

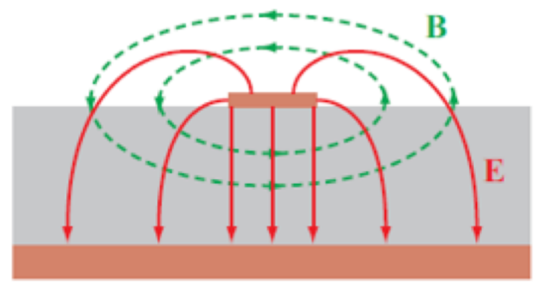

Figure 2 Cross-sectional view with E and B field lines

Figure 1: Microstrip line geometry and cross-sectional view with E and B field lines

$$u_p = \frac{c}{\sqrt{\varepsilon_{\text{eff}}}},$$

with c being the speed of light in free space.

The assignment provides curve-fit formulas that relate ε_{eff} and the characteristic impedance Z_0 to the width-to-thickness ratio $s = w/h$. First, ε_{eff} is computed as

$$\boxed{\varepsilon_{\text{eff}} = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left(1 + \frac{10}{s}\right)^{-xy}}$$

where the intermediate parameters x and y depend on material and geometry:

$$\boxed{x = 0.56\left(\frac{\varepsilon_r - 0.9}{\varepsilon_r + 3}\right)^{0.05} \quad \text{and} \quad y = 1 + 0.02\ln\left(\frac{s^4 + 0.00037\, s^2}{s^4 + 0.43}\right) + 0.05\ln(1 + 0.00017\, s^3).}$$

Given ε_{eff}, the characteristic impedance is then obtained from

$$\boxed{Z_0 = \frac{60}{\sqrt{\varepsilon_{\text{eff}}}} \ln\left(\frac{6 + (2\pi - 6)\, e^{-t}}{s} + \sqrt{1 + \frac{4}{s^2}}\right), \quad t = (\frac{30.67}{s})^{0.75}.}$$

These expressions provide a method to compute the characteristic impedance which is dependent on the ratio of s.

In practical design, however, we often start from a desired impedance Z_0 and a chosen ε_r and must determine the required $s = w/h$. Reverse engineering of this equation is non-trivial, so the assignment provides two inverse approximations for s, each with a specific validity constraint on Z_0:

(a) For $Z_0 \leq (44 - 2\varepsilon_r)\ \Omega$**:**

$$\boxed{s = \frac{w}{h} = \frac{2}{\pi}\left[(q-1) - \ln(2q-1) + \frac{\varepsilon_r - 1}{2\varepsilon_r}\left(\ln(q-1) + 0.29 - \frac{0.52}{\varepsilon_r}\right)\right], \qquad q = \frac{60\pi^2}{Z_0\sqrt{\varepsilon_r}}.}$$

(b) For $Z_0 \geq (44 - 2\varepsilon_r)\ \Omega$**:**

$$\boxed{s = \frac{w}{h} = \frac{8e^{\,p}}{e^{2p} - 2}, p = \sqrt{\frac{\varepsilon_r + 1}{2}}(\frac{Z_0}{60}) + (\frac{\varepsilon_r - 1}{\varepsilon_r + 1})(0.23 + \frac{0.12}{\varepsilon_r}).}$$

Together, these four equations and the two constrained inverse formulas create the basis used throughout the assignment. The forward model (s given) predicts how geometry and dielectric constant set the propagation. The inverse relation allows us to select a practical height and width for a required impedance. The constraint $Z_0 \lesseqgtr (44 - 2\varepsilon_r)\ \Omega$ determines which constraint expression should be applied. It is important to note that the value of the characteristic impedance for each approximation is only accurate if it lies within about 2% over its intended range.

Goal of the assignment:
Compute Z_0 with the forward model across $s =$ 0.5-5 (fixed $h = 1mm$ and $\varepsilon_r = 10$ with s in increments of 0.1), compare these values to a reference to the simulation at s=0.5, 1, 1.2, 1.5, 2, 3, 4, 5 and plot the relative difference (as a percent) of Z_0 versus s. Repeat the calculations without the plot for $\varepsilon_r = 2$. Then, using ε_r and the computed Z_0, reverse-engineer s with approximations (a) and (b) and plot the relative difference (as a percent) of s versus Z_0. From these plots, identify the range of Z_0 for where each approximation formula meets the 2% criteria and discuss whether the observed accuracy aligns with the theory's stated constraints.

Flow Diagram:

Part 1 – Relative difference of Z_0 vs. s

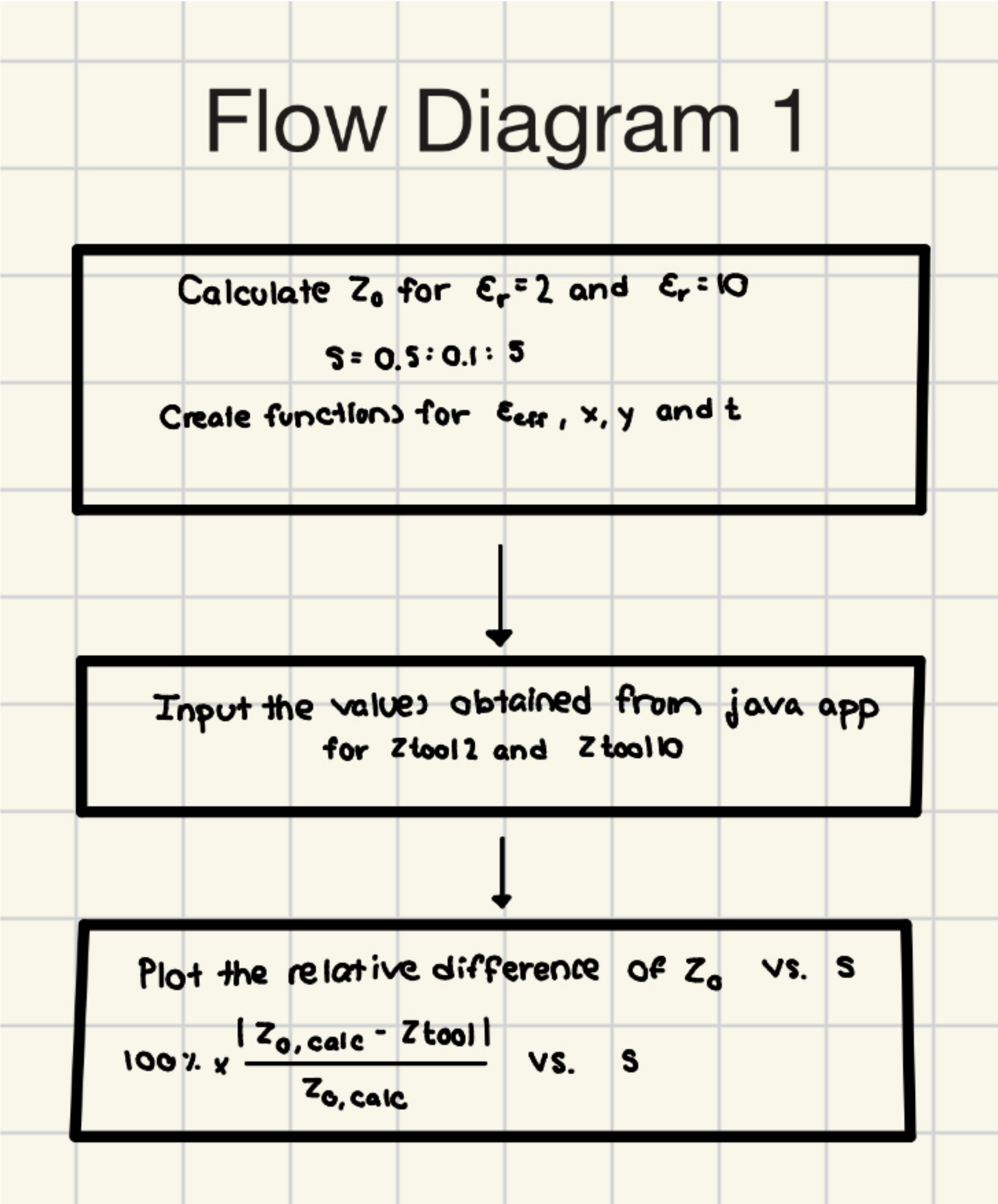

Figure 1: Flow diagram of the relative difference of Z_0 as a function of s

Part 2 – Relative difference of s vs. Z_0

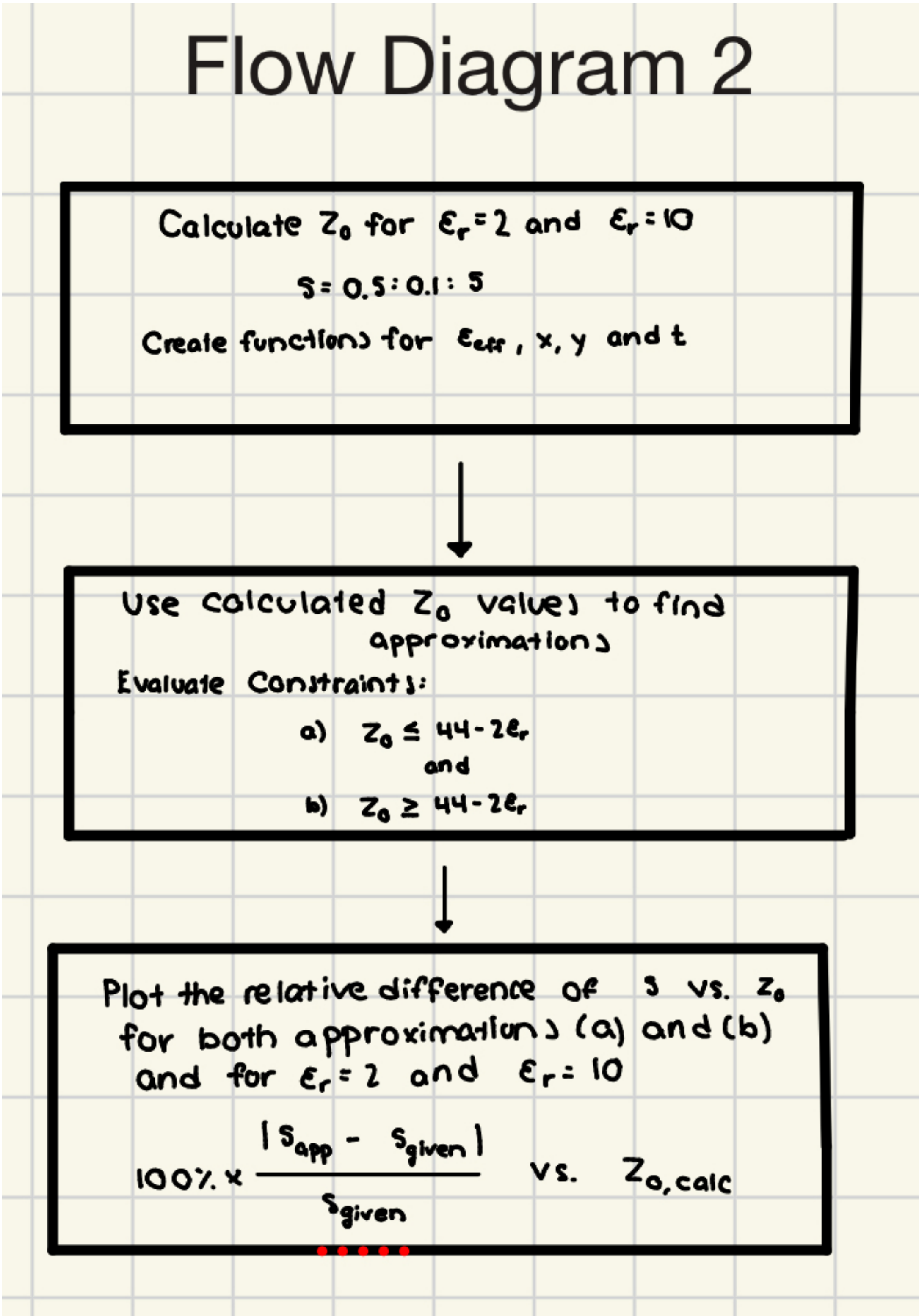

Figure 2: Flow diagram of the relative difference of s as a function of Z_0

Code:

Part 1 – Relative difference of Z_0 vs. s

```matlab
clc; clear; close all;

s=0.5:0.1:5;

for i = 1:numel(s) %line impedance values for epslion_r=2
    epsilon_r=2;
    x=0.56.*((epsilon_r-0.9)./(epsilon_r+3)).^0.05;                          %x used to calculate epsion_eff
    y=1+0.02.*log((s.^4+0.00037.*s.^2)./(s.^4+0.43))+0.05.*log(1+0.00017.*s.^3);  %y used to calculate epsilon_Eff
    epsilon_eff=(epsilon_r+1)./2+((epsilon_r-1)./2).*(1+10./s).^(-x.*y);    %effective permittivity
    t=(30.67./s).^0.75;                                                          %t used to calculate impedance_char
    impedance_char_2=(60./sqrt(epsilon_eff)).*log((6+(2.*pi-6).*exp(-t))./s+sqrt(1+4./s.^2)); %line impedance
end

for j = 1:numel(s) %line impedance values for epslion_r=10
    epsilon_r=10;
    x=0.56.*((epsilon_r-0.9)./(epsilon_r+3)).^0.05;                          %x used to calculate epsion_eff
    y=1+0.02.*log((s.^4+0.00037.*s.^2)./(s.^4+0.43))+0.05.*log(1+0.00017.*s.^3);  %y used to calculate epsilon_Eff
    epsilon_eff=(epsilon_r+1)./2+((epsilon_r-1)./2).*(1+10./s).^(-x.*y);    %effective permittivity
    t=(30.67./s).^0.75;                                                          %t used to calculate impedance_char
    impedance_char_10=(60./sqrt(epsilon_eff)).*log((6+(2.*pi-6).*exp(-t))./s+sqrt(1+4./s.^2)); %line impedance
end

% -------------------- Data from screenshots --------------------
s_er2   = [0.5 1.0 1.2 1.5 2.0 3.0 4.0 5.0]';
Ztool2 = [131.674247, 98.902177, 90.513193, 80.812594, 68.767620, 53.287893, 43.677636, 37.088415];%ohms

s_er10  = [0.5 1.0 1.2 1.5 2.0 3.0 4.0 5.0]';
Ztool10 = [65.694609 48.719215 44.391902 39.411747 33.281708 25.497875 20.725735 17.484552]; % ohms

% requested sample points
s_req = [0.5 1 1.2 1.5 2 3 4 5];

% calculated Z0 at those points (avoids index issues)
```

Figure 3: Code for the relative difference of Z_0 as a function of s – (1 of 2)

```matlab
% calculated Z0 at those points (avoids index issues)
Zcalc_pts = interp1(s, impedance_char_10, s_req, 'linear');

% simulated Z0 from screenshots
Zsim_pts  = Ztool10;

% relative difference (%)
rel_diff_pct = 100 * abs(Zsim_pts - Zcalc_pts) ./ Zsim_pts;

% plot
figure('Color','w'); hold on; grid on; box on;
plot(s_req, rel_diff_pct, '-o', 'LineWidth', 1.8, 'MarkerSize', 6);
xlabel('s = w/h'); ylabel('Relative difference in Z_0  [%]');
title('|\Delta Z_0| / Z_{0,sim} × 100,  \epsilon_r = 10');
yline(0, ':');
```

Figure 4: Code for the relative difference of Z_0 as a function of s – (2 of 2)

Part 2 – Relative difference of s vs. Z_0

```
clc; clear; close all;

s=0.5:0.1:5;

for i = 1:numel(s) %line impedance values for epslion_r=2
    epsilon_r=2;
    x=0.56.*((epsilon_r-0.9)./(epsilon_r+3)).^0.05;                         %x used to calculate epsion_eff
    y=1+0.02.*log((s.^4+0.00037.*s.^2)./(s.^4+0.43))+0.05.*log(1+0.00017.*s.^3);  %y used to calculate epsilon_Eff
    epsilon_eff=(epsilon_r+1)./2+((epsilon_r-1)./2).*(1+10./s).^(-x.*y);    %effective permittivity
    t=(30.67./s).^0.75;                                                         %t used to calculate impedance_char
    impedance_char_2=(60./sqrt(epsilon_eff)).*log((6+(2.*pi-6).*exp(-t))./s+sqrt(1+4./s.^2)); %line impedance
end

for j = 1:numel(s) %line impedance values for epslion_r=10
    epsilon_r=10;
    x=0.56.*((epsilon_r-0.9)./(epsilon_r+3)).^0.05;                         %x used to calculate epsion_eff
    y=1+0.02.*log((s.^4+0.00037.*s.^2)./(s.^4+0.43))+0.05.*log(1+0.00017.*s.^3);  %y used to calculate epsilon_Eff
    epsilon_eff=(epsilon_r+1)./2+((epsilon_r-1)./2).*(1+10./s).^(-x.*y);    %effective permittivity
    t=(30.67./s).^0.75;                                                         %t used to calculate impedance_char
    impedance_char_10=(60./sqrt(epsilon_eff)).*log((6+(2.*pi-6).*exp(-t))./s+sqrt(1+4./s.^2)); %line impedance
end

s_from_Z0_a = @(Z0,er) (2/pi) .* ( ...
    ( (60*pi^2) ./ (Z0.*sqrt(er)) - 1 ) ...
    - log( 2*(60*pi^2)./(Z0.*sqrt(er)) - 1 ) ...
    + ((er-1)./(2*er)) .* ( log( (60*pi^2)./(Z0.*sqrt(er)) - 1 ) + 0.29 - 0.52./er ) );

s_from_Z0_b = @(Z0,er) ( 8 .* exp( (Z0/60).*sqrt((er+1)/2) + ((er-1)./(er+1)).*(0.23 + 0.12./er) ) ) ...
                     ./ ( exp( 2*((Z0/60).*sqrt((er+1)/2) + ((er-1)./(er+1)).*(0.23 + 0.12./er)) ) - 2 );

% (small numeric guard for logs when q≈1)
guard = @(x) max(x, 1+eps);

% Wrapped version that guards q = 60π²/(Z0√εr)
s_from_Z0_a_guarded = @(Z0,er) (2/pi) .* ( ...
```

Figure 5: Code for the relative difference of s as a function of Z_0 – (1 of 3)

```
s_from_Z0_a_guarded = @(Z0,er) (2/pi) .* ( ...
    ( (60*pi^2)./(Z0.*sqrt(er)) - 1 ) ...
    - log( 2*guard((60*pi^2)./(Z0.*sqrt(er))) - 1 ) ...
    + ((er-1)./(2*er)) .* ( log( guard((60*pi^2)./(Z0.*sqrt(er))) - 1 ) + 0.29 - 0.52./er ) );

%% ================= Step 1: compute s-approximations & relative diffs =====
% eps_r = 2
er2   = 2;
Z2    = impedance_char_2(:).';      % row
sA_2  = s_from_Z0_a_guarded(Z2,er2);
sB_2  = s_from_Z0_b(Z2,er2);
relA_2 = 100*abs(sA_2 - s)./s;      % percent difference vs given s
relB_2 = 100*abs(sB_2 - s)./s;

% eps_r = 10
er10  = 10;
Z10   = impedance_char_10(:).';     % row
sA_10 = s_from_Z0_a_guarded(Z10,er10);
sB_10 = s_from_Z0_b(Z10,er10);
relA_10 = 100*abs(sA_10 - s)./s;
relB_10 = 100*abs(sB_10 - s)./s;

%% ================= Step 2: two plots (one per eps_r, two curves each) ====
thr2  = 44 - 2*er2;     % region split in the handbook formula
thr10 = 44 - 2*er10;

% ---- Figure for eps_r = 2 ----
figure('Color','w');
plot(Z2, relA_2, '-', 'LineWidth', 2); hold on;
plot(Z2, relB_2, '--', 'LineWidth', 2);
xline(thr2, ':', sprintf('Z_0 = 44-2\\epsilon_r = %.1f\\Omega', thr2), ...
    'LabelVerticalAlignment','bottom','LabelOrientation','horizontal');
grid on; box on;
xlabel('Z_0 [\Omega]');
ylabel('Relative difference in s  [%]');
```

Figure 6: Code for the relative difference of s as a function of Z_0 – (2 of 3)

```
title('Relative difference of s vs Z_0   (\epsilon_r = 2)');
legend('Approx. (a)', 'Approx. (b)', 'Location','northeast');

% ---- Figure for eps_r = 10 ----
figure('Color','w');
plot(Z10, relA_10, '-', 'LineWidth', 2); hold on;
plot(Z10, relB_10, '--', 'LineWidth', 2);
xline(thr10, ':', sprintf('Z_0 = 44-2\\epsilon_r = %.1f\\Omega', thr10), ...
    'LabelVerticalAlignment','bottom','LabelOrientation','horizontal');
grid on; box on;
xlabel('Z_0 [\Omega]');
ylabel('Relative difference in s  [%]');
title('Relative difference of s vs Z_0   (\epsilon_r = 10)');
legend('Approx. (a)', 'Approx. (b)', 'Location','northeast');
```

Figure 7: Code for the relative difference of s as a function of Z_0 – (3 of 3)

Virtual Simulations:

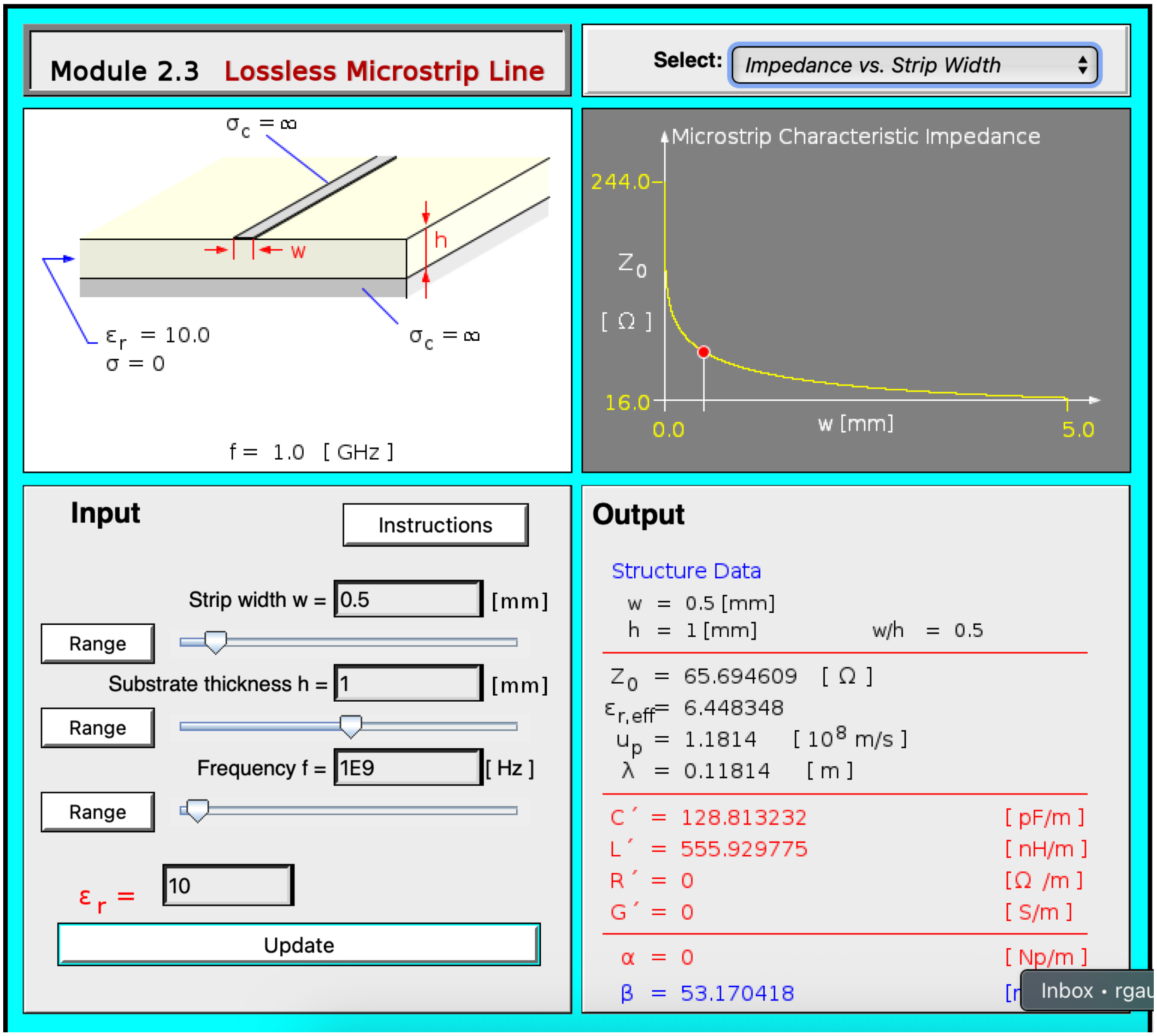

Figure 8: Virtual simulation for $\varepsilon_r = 10$ and $s = 0.5$

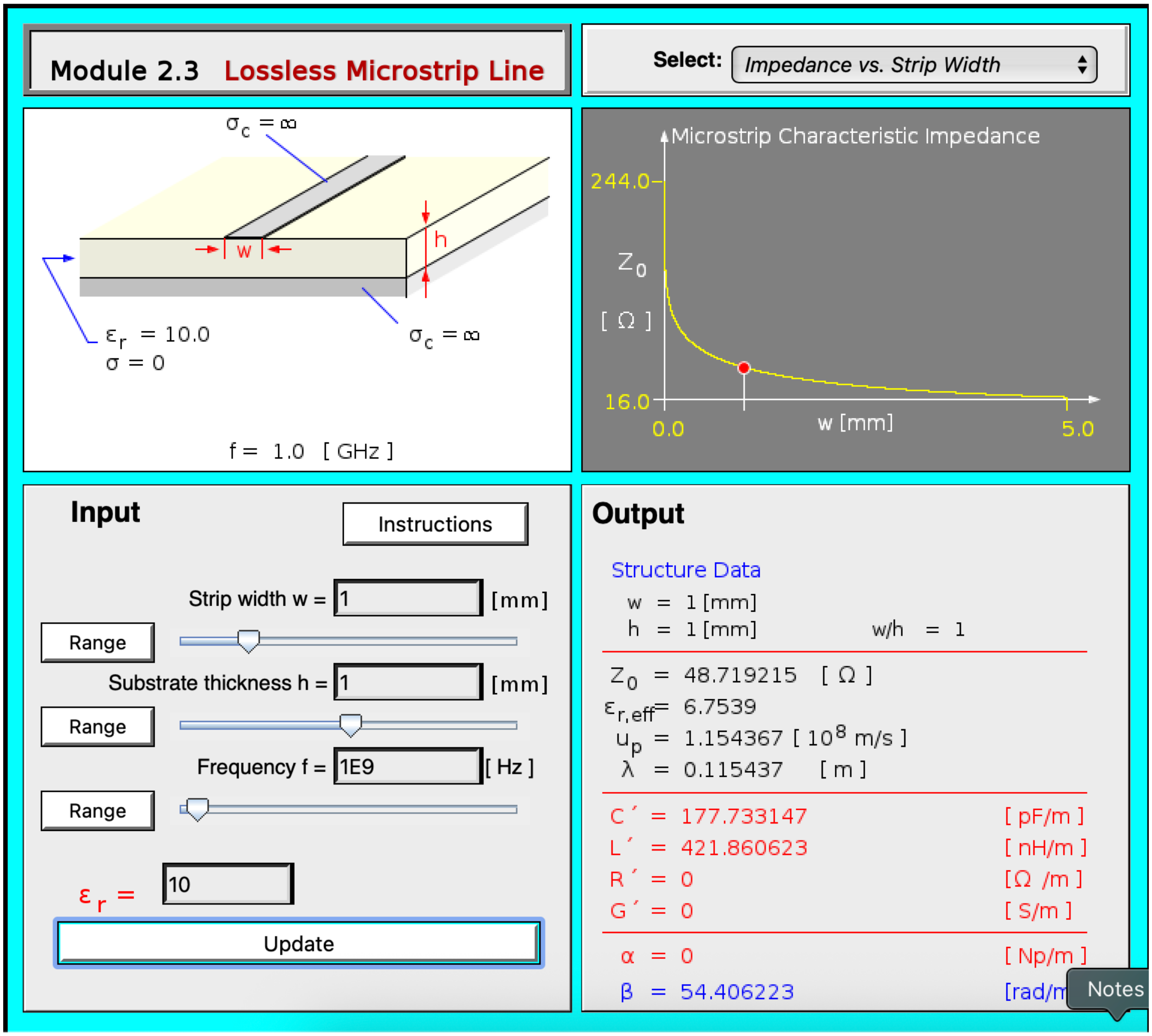

Figure 9: Virtual simulation for $\varepsilon_r = 10$ and $s = 1$

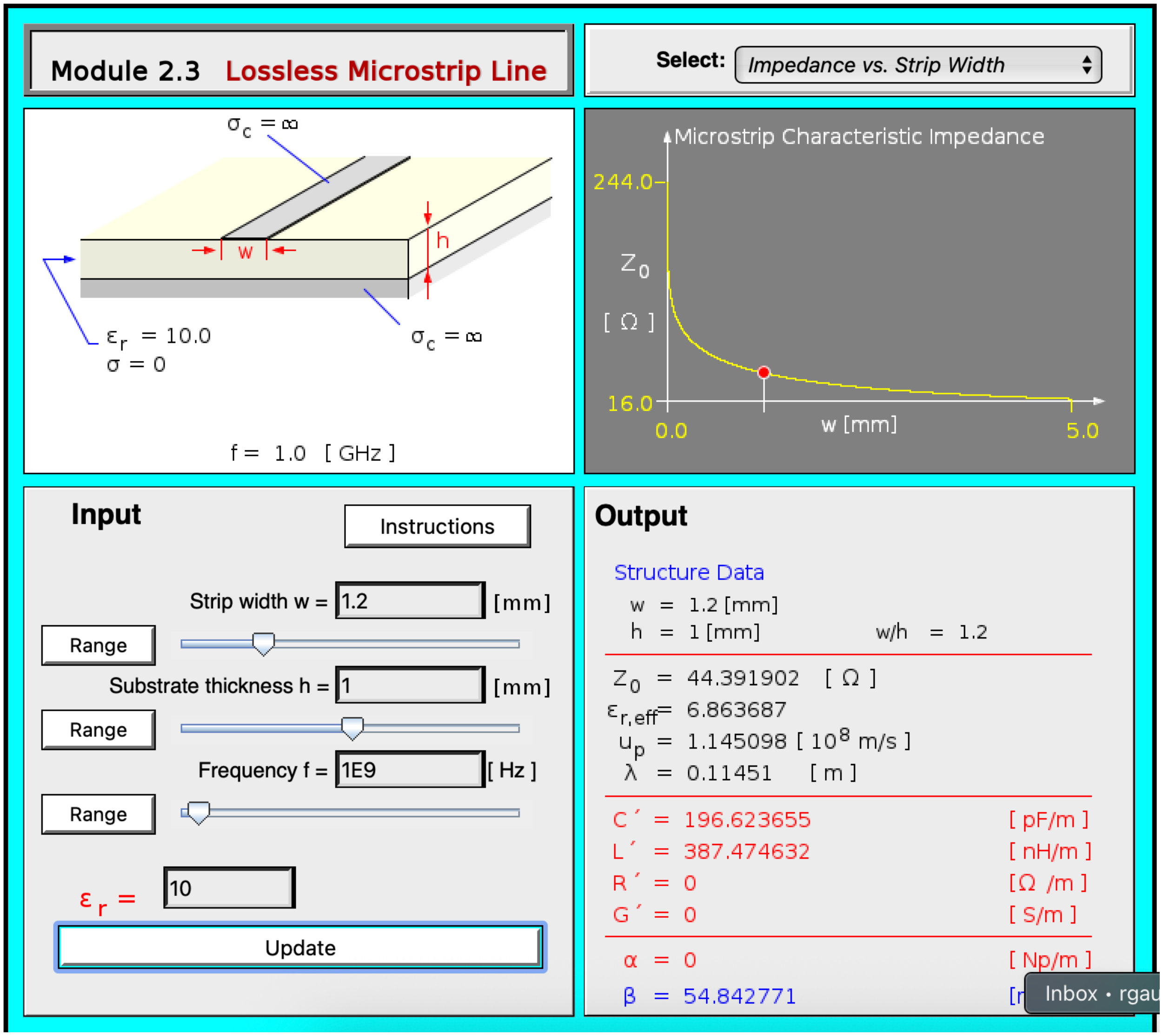

Figure 10: Virtual simulation for $\varepsilon_r = 10$ and $s = 1.2$

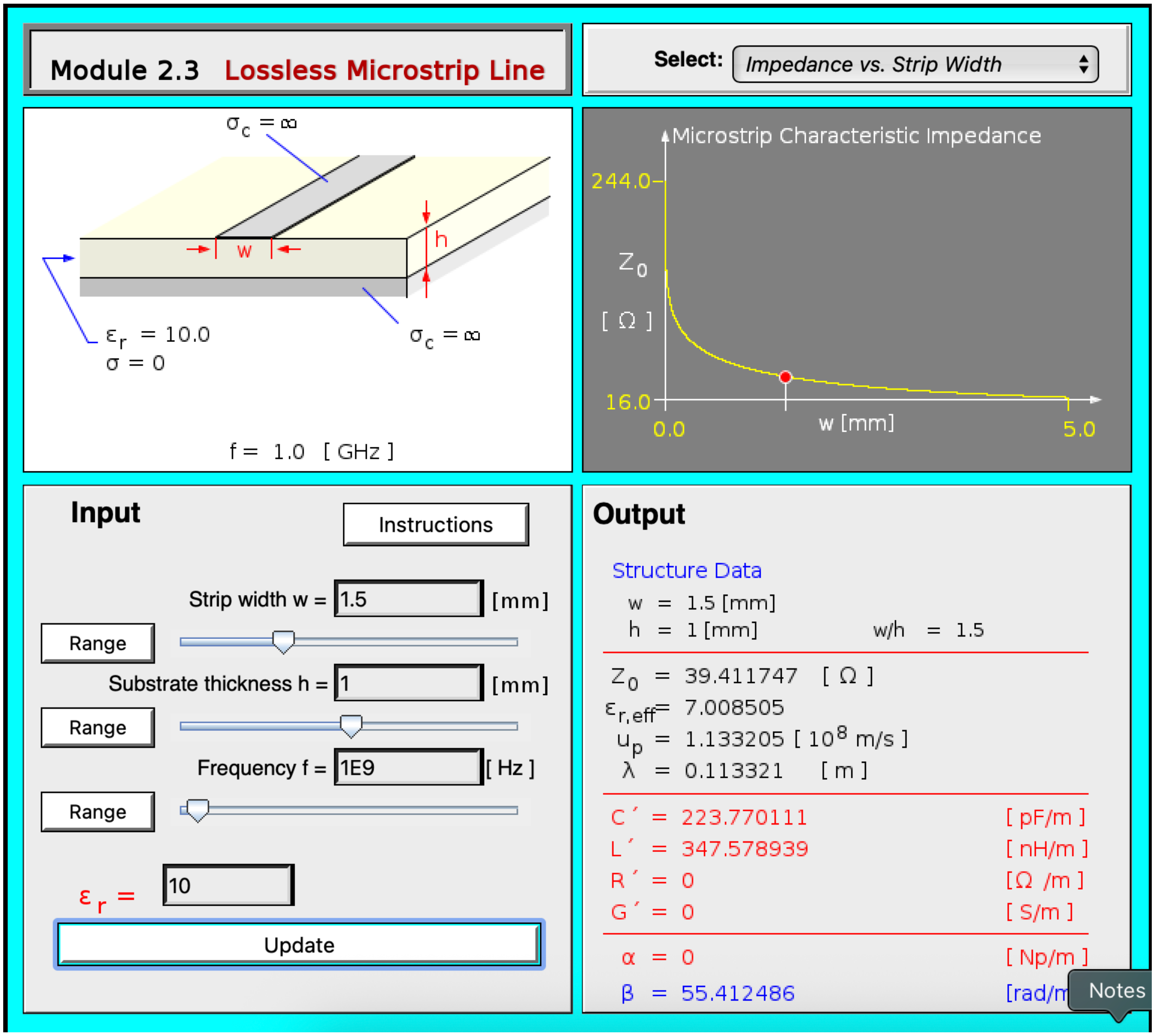

Figure 11: Virtual simulation for $\varepsilon_r = 10$ and $s = 1.5$

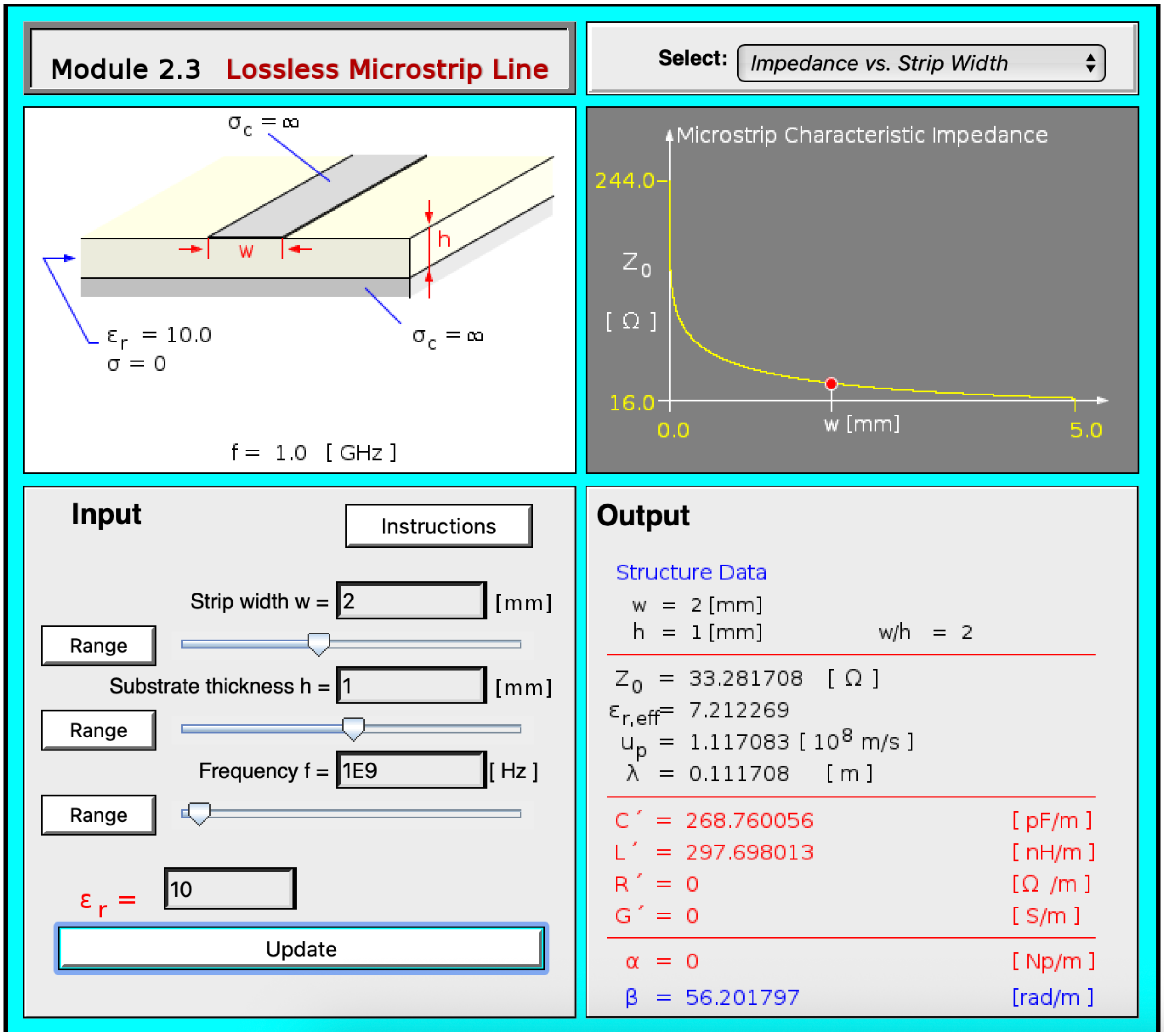

Figure 12: Virtual simulation for $\varepsilon_r = 10$ and $s = 2$

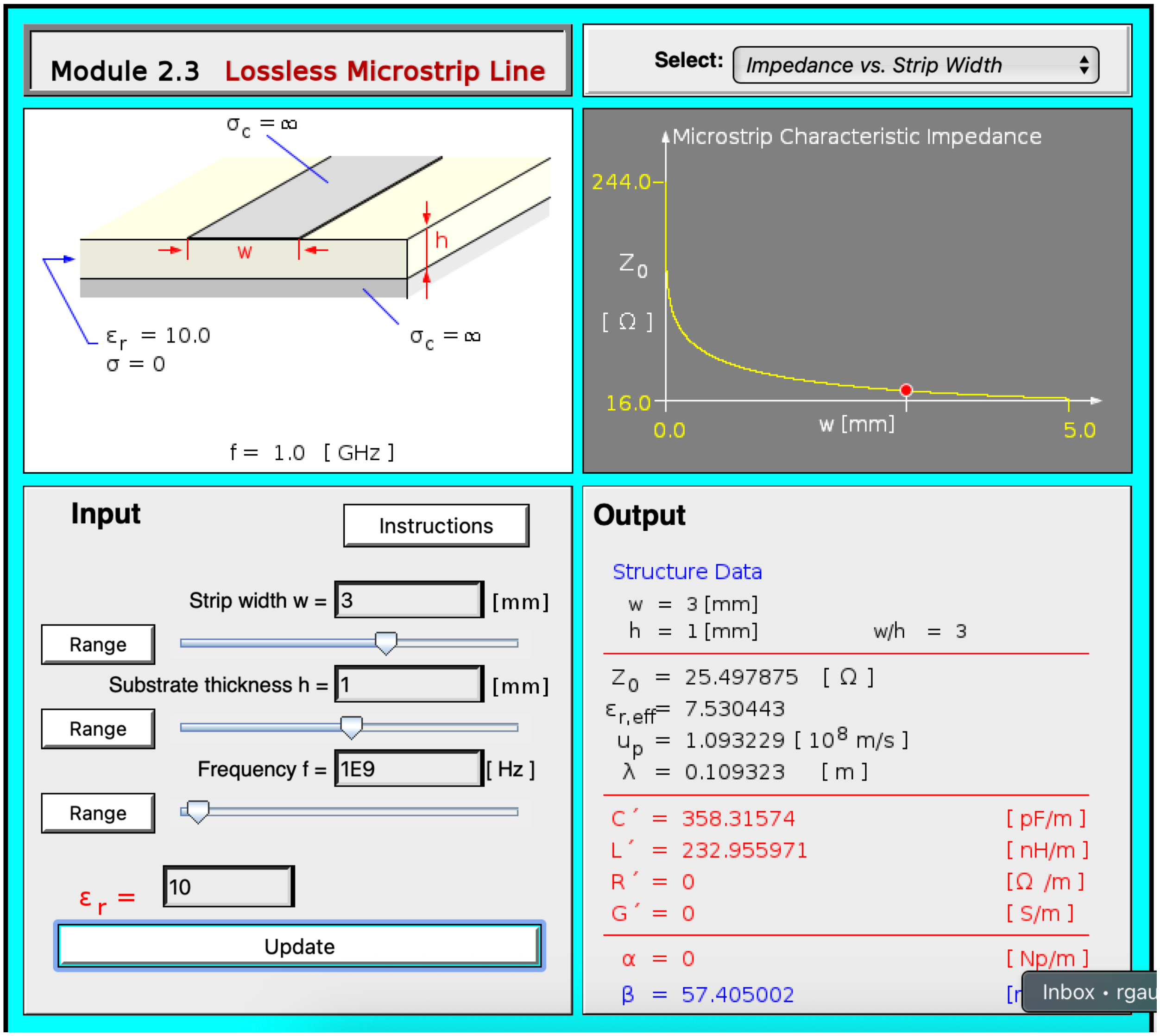

Figure 13: Virtual simulation for $\varepsilon_r = 10$ and $s = 3$

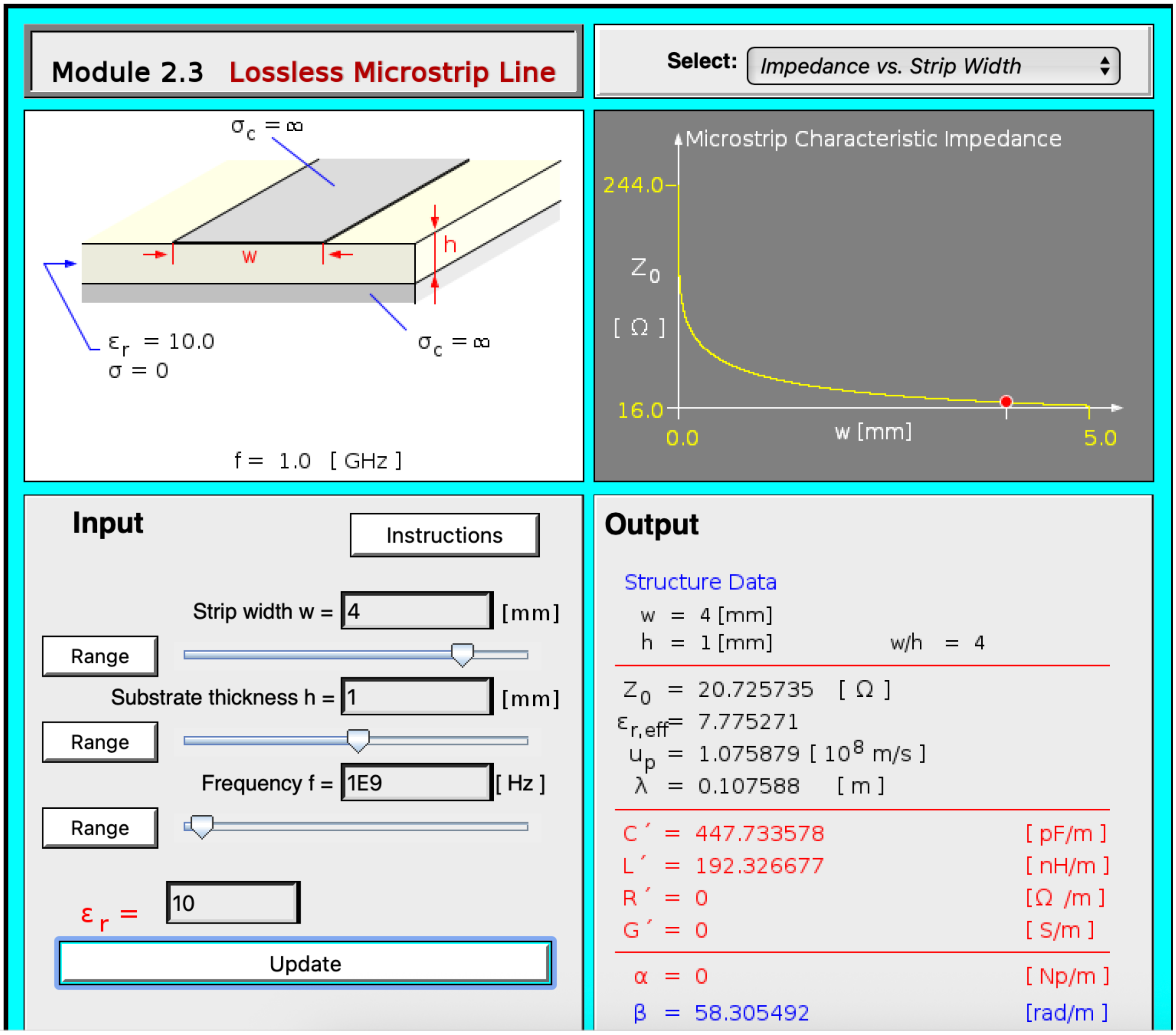

Figure 14: Virtual simulation for $\varepsilon_r = 10$ and $s = 4$

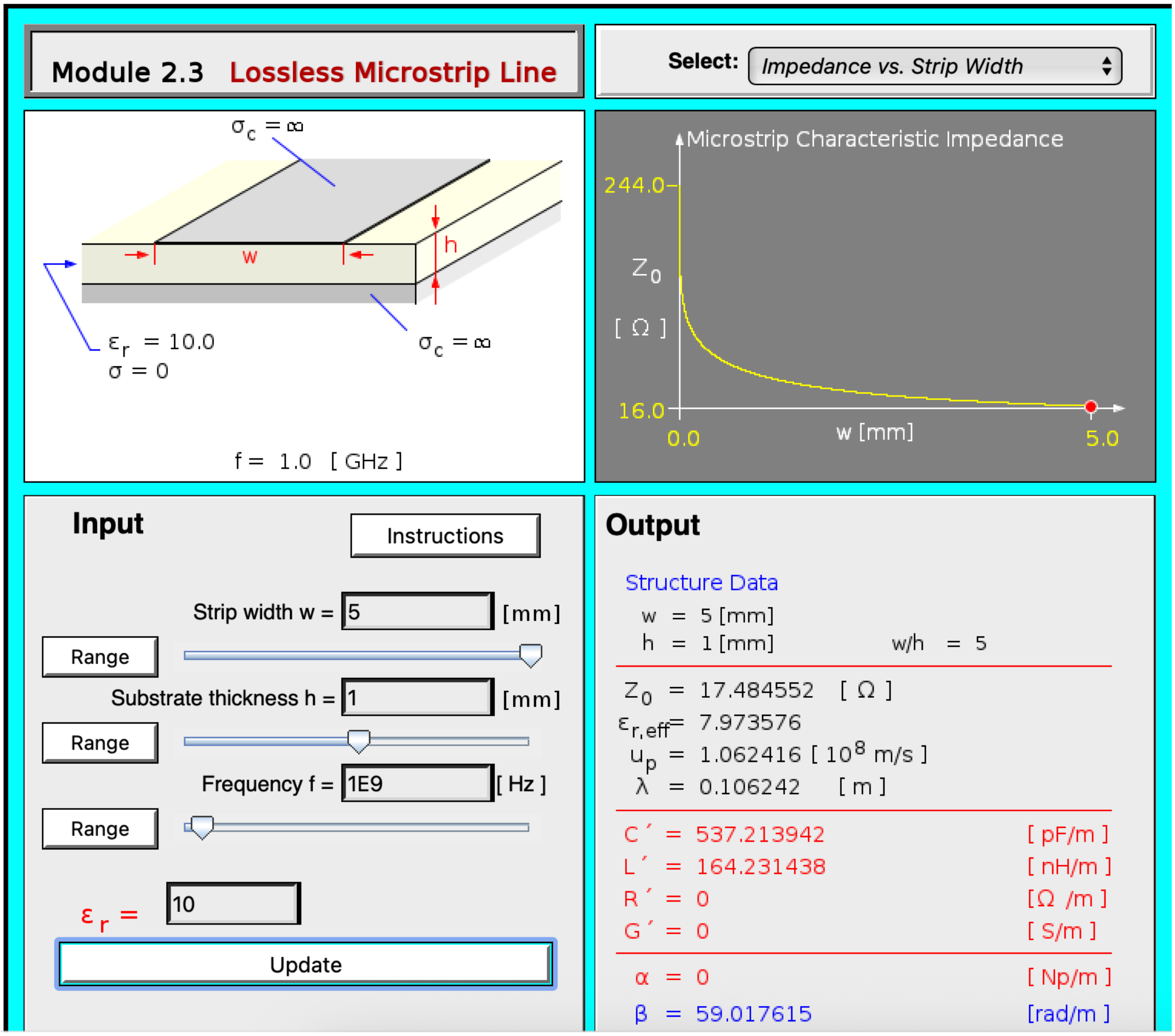

Figure 15: Virtual simulation for $\varepsilon_r = 10$ and $s = 5$

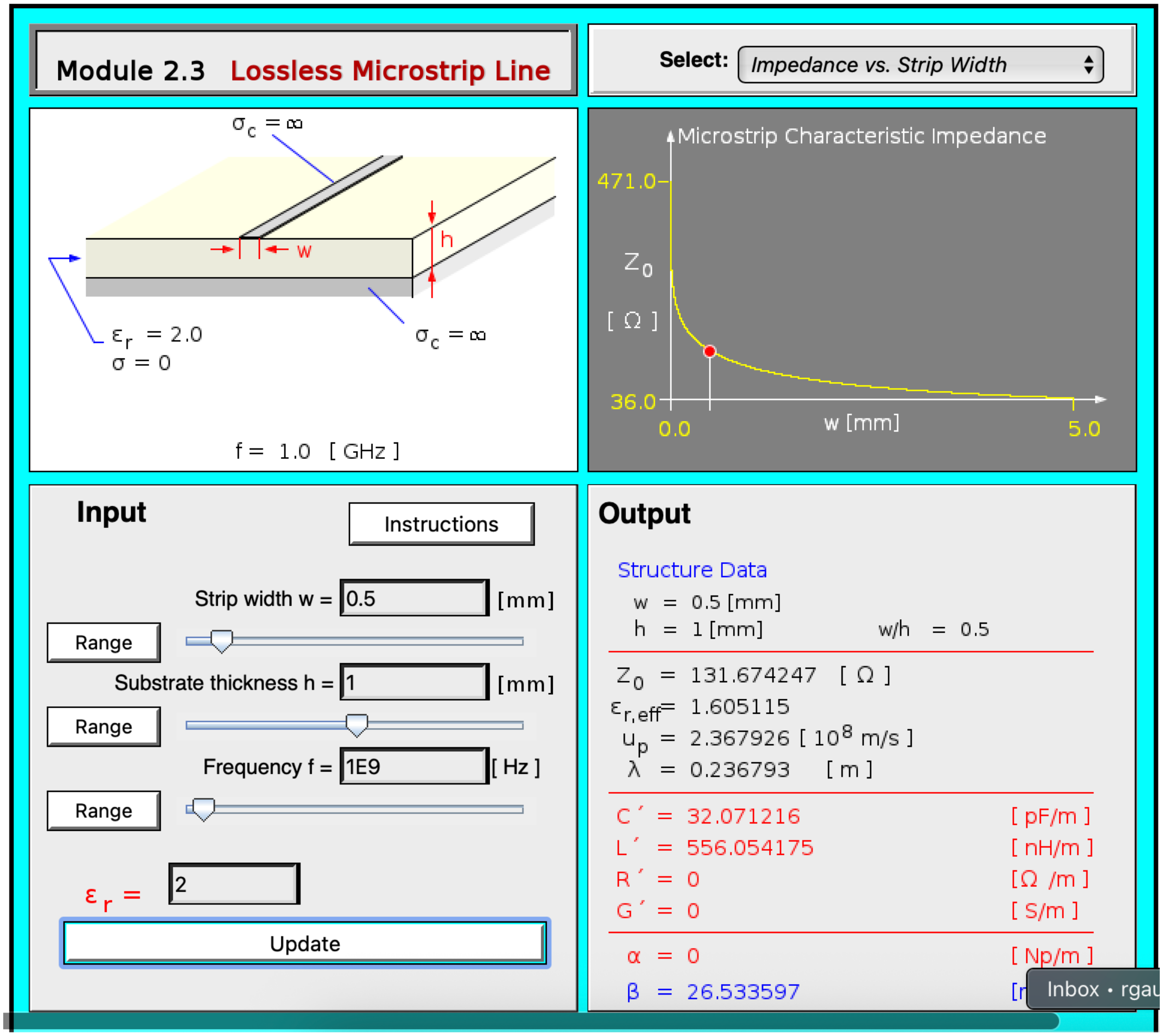

Figure 16: Virtual simulation for $\varepsilon_r = 2$ and $s = 0.5$

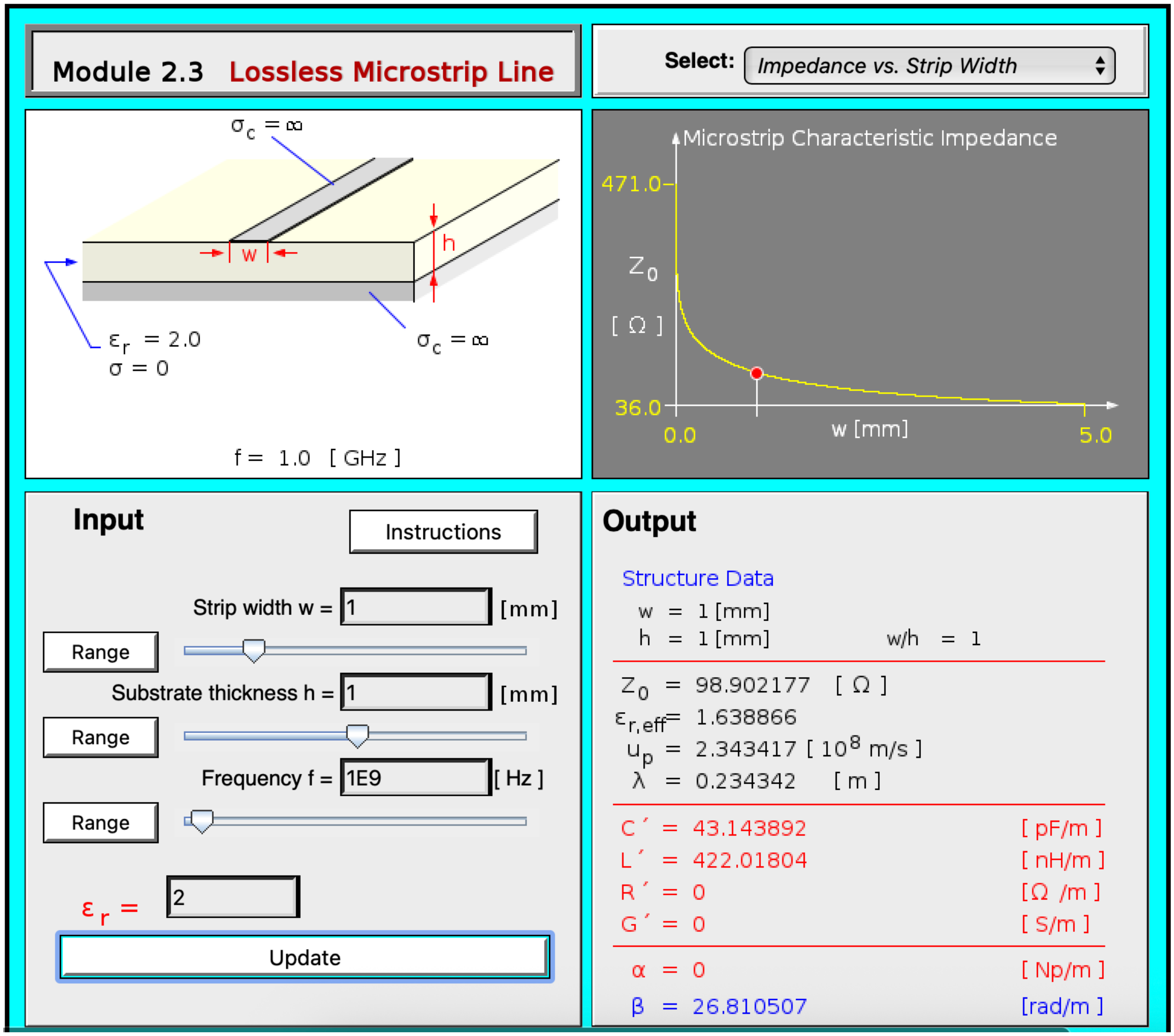

Figure 17: Virtual simulation for $\varepsilon_r = 2$ and $s = 1$

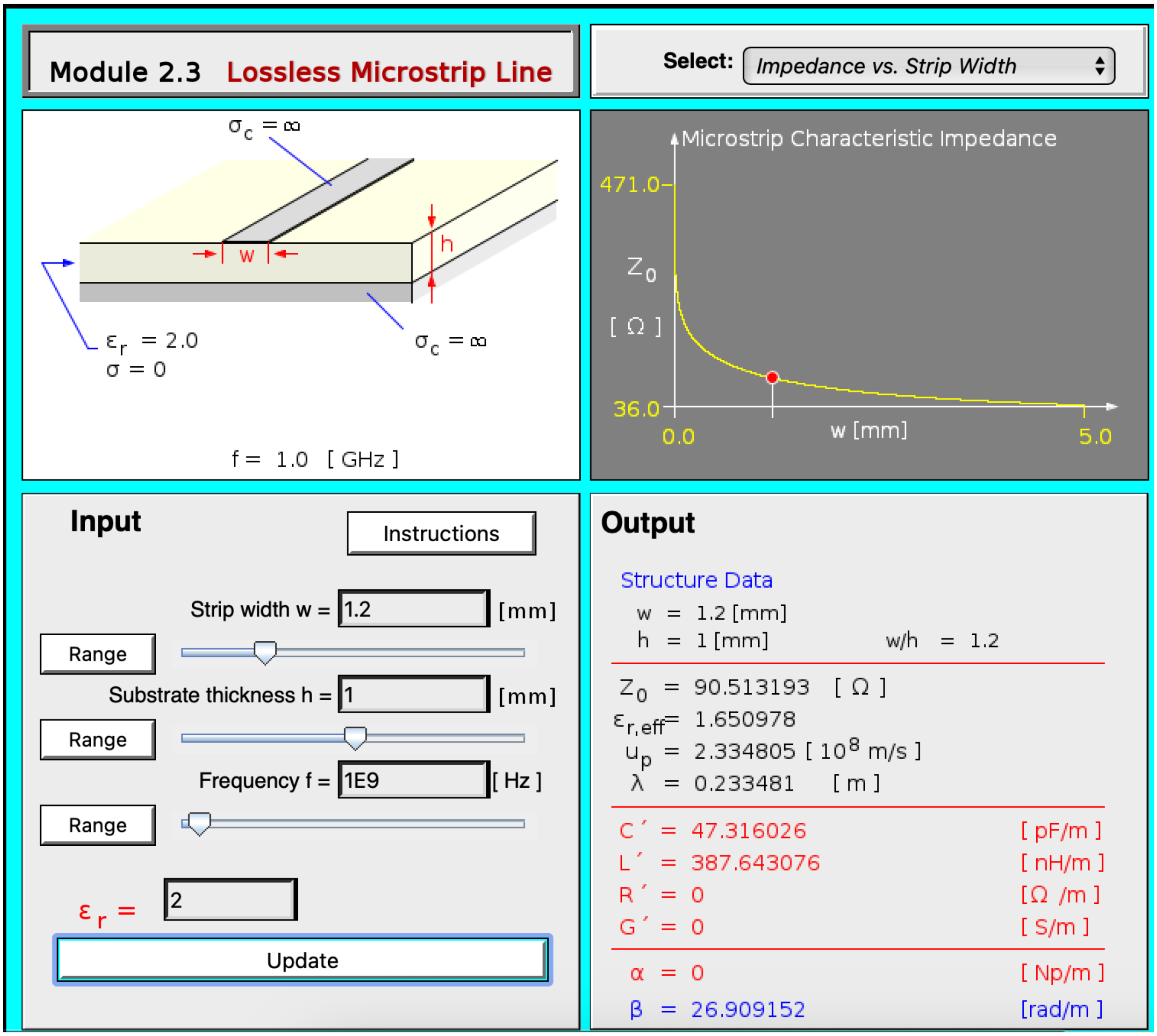

Figure 18: Virtual simulation for $\varepsilon_r = 2$ and $s = 1.2$

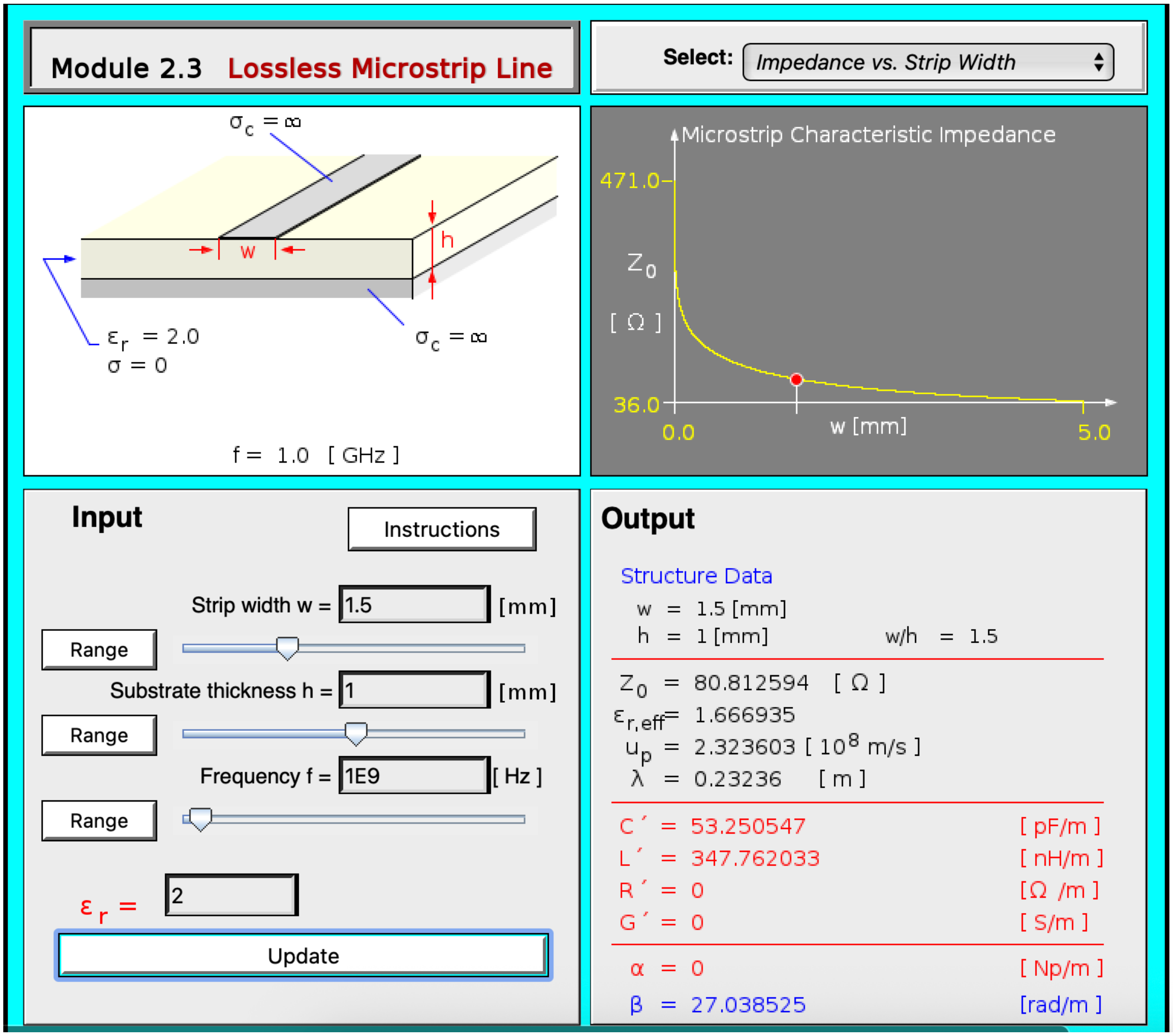

Figure 19: Virtual simulation for $\varepsilon_r = 2$ and $s = 1.5$

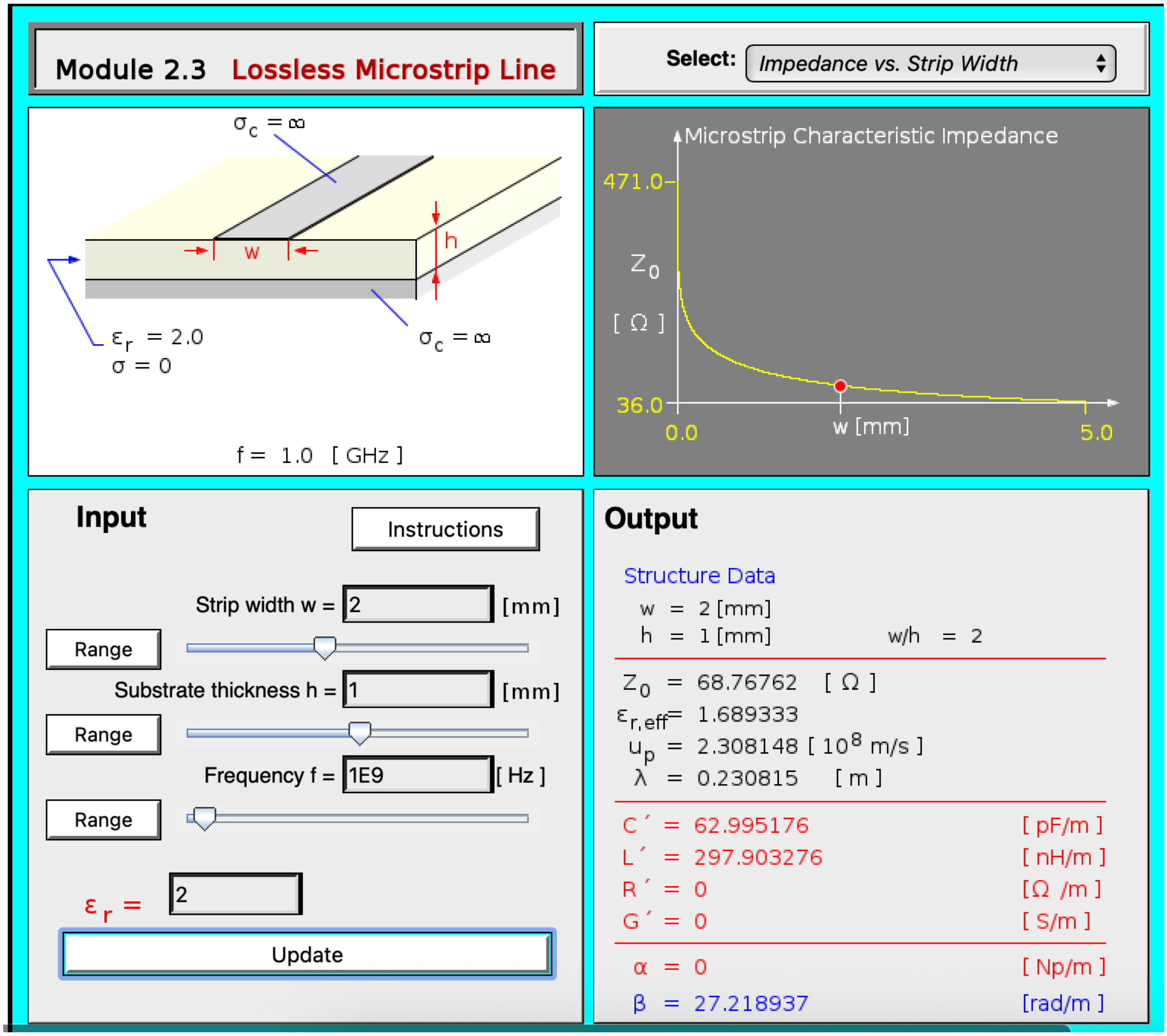

Figure 20: Virtual simulation for $\varepsilon_r = 2$ and $s = 2$

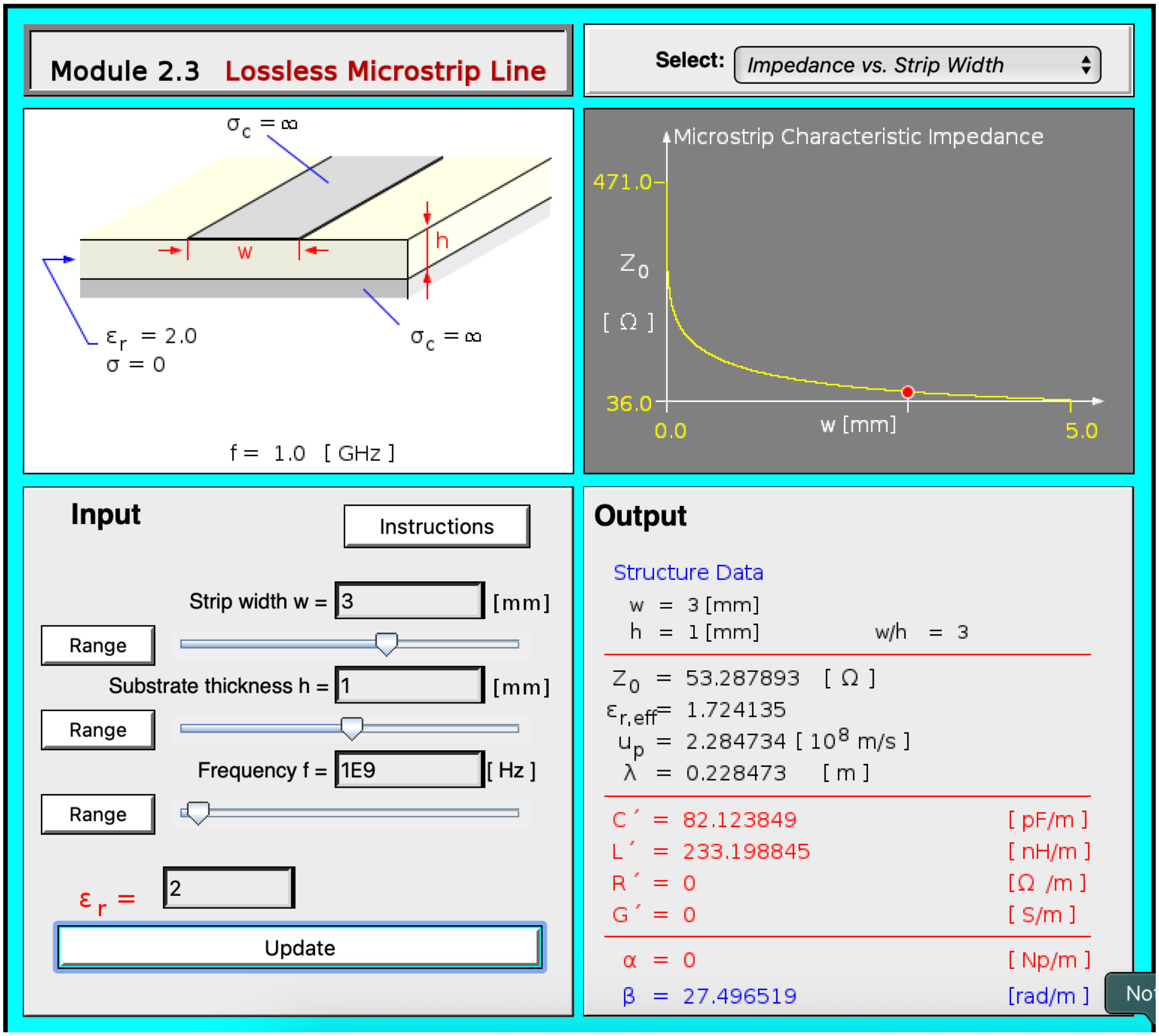

Figure 21: Virtual simulation for $\varepsilon_r = 2$ and $s = 3$

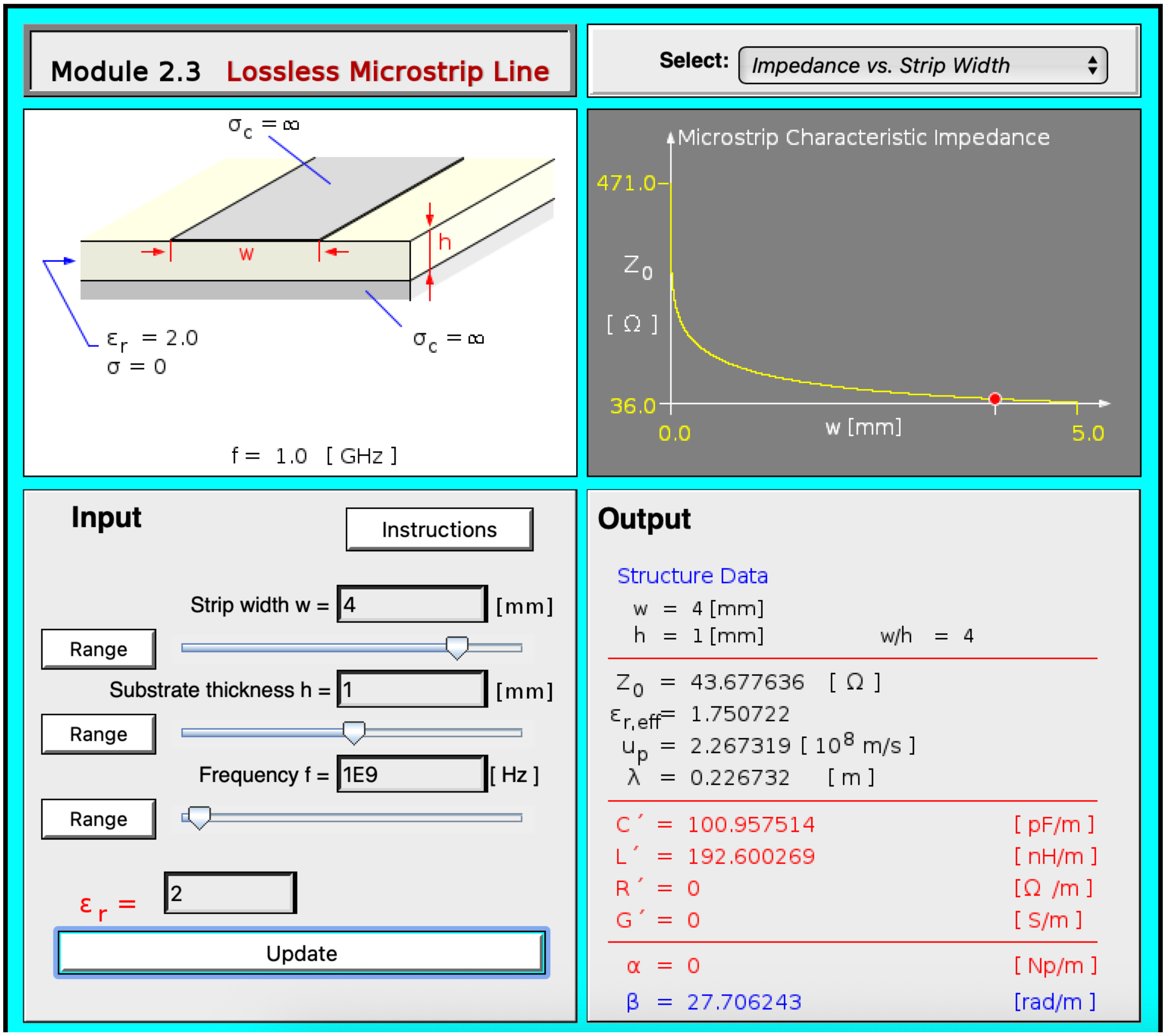

Figure 22: Virtual simulation for $\varepsilon_r = 2$ and $s = 4$

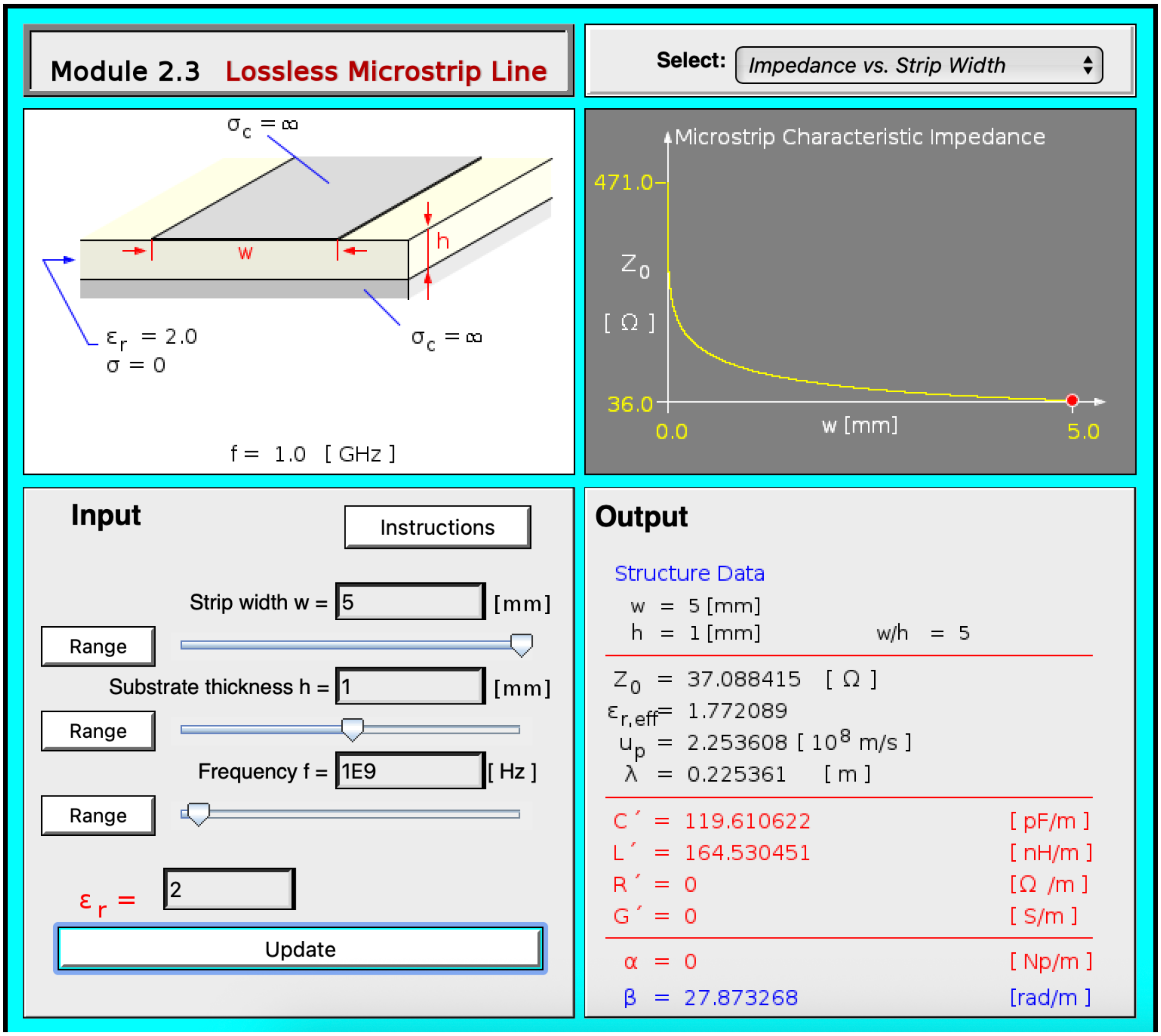

Figure 23: Virtual simulation for $\varepsilon_r = 2$ and $s = 5$

Table 1: Simulated results for $\varepsilon_r = 10$

s	0.5	1.0	1.2	1.5	2.0	3.0	4.0	5.0
Z_0	65.922	48.822	44.511	39.421	33.248	25.484	20.738	17.513

Table 2: Simulated results for $\varepsilon_r = 2$

s	0.5	1.0	1.2	1.5	2.0	3.0	4.0	5.0
Z_0	131.475	98.631	90.311	80.441	68.383	53.037	43.529	37.003

Results:

Part 1 – Relative difference of Z_0 vs. s

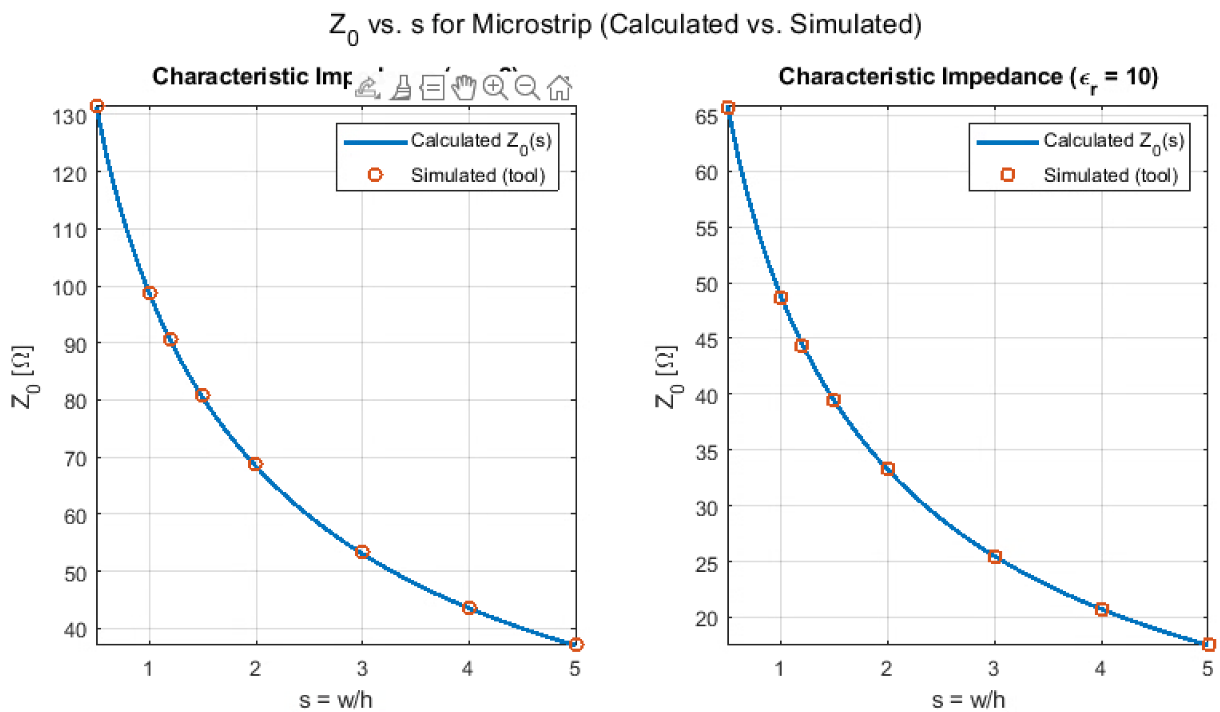

Figure 24: Comparison of Z_0 vs. s for Part 1

s_w_over_h	Z0_eps2_ohm	Z0_eps10_ohm
0.5	131.47	65.922
0.6	122.71	61.354
0.7	115.36	57.522
0.8	109.04	54.233
0.9	103.52	51.362
1	98.631	48.822
1.1	94.257	46.554
1.2	90.311	44.512
1.3	86.727	42.66
1.4	83.451	40.97
1.5	80.441	39.421
1.6	77.664	37.995
1.7	75.092	36.676
1.8	72.7	35.451
1.9	70.47	34.312
2	68.383	33.248
2.1	66.427	32.252
2.2	64.588	31.317
2.3	62.855	30.438
2.4	61.219	29.609

Figure 25: Calculated values of Z_0 for $\varepsilon_r = 10$ and $\varepsilon_r = 2$ – (1 of 3)

2.5	59.673	28.827
2.6	58.207	28.086
2.7	56.816	27.385
2.8	55.494	26.719
2.9	54.236	26.086
3	53.037	25.484
3.1	51.892	24.91
3.2	50.799	24.362
3.3	49.753	23.838
3.4	48.751	23.338
3.5	47.791	22.859
3.6	46.87	22.399
3.7	45.985	21.958
3.8	45.135	21.535
3.9	44.317	21.129
4	43.529	20.738
4.1	42.77	20.361
4.2	42.038	19.998
4.3	41.332	19.649
4.4	40.65	19.311
4.5	39.991	18.986
4.6	39.354	18.671
4.7	38.738	18.367
4.8	38.141	18.073

Figure 26: Calculated values of Z_0 for $\varepsilon_r = 10$ and $\varepsilon_r = 2$ – (2 of 3)

4.9	37.563	17.789
5	37.003	17.513

Figure 27: Calculated values of Z_0 for $\varepsilon_r = 10$ and $\varepsilon_r = 2$ – (3 of 3)

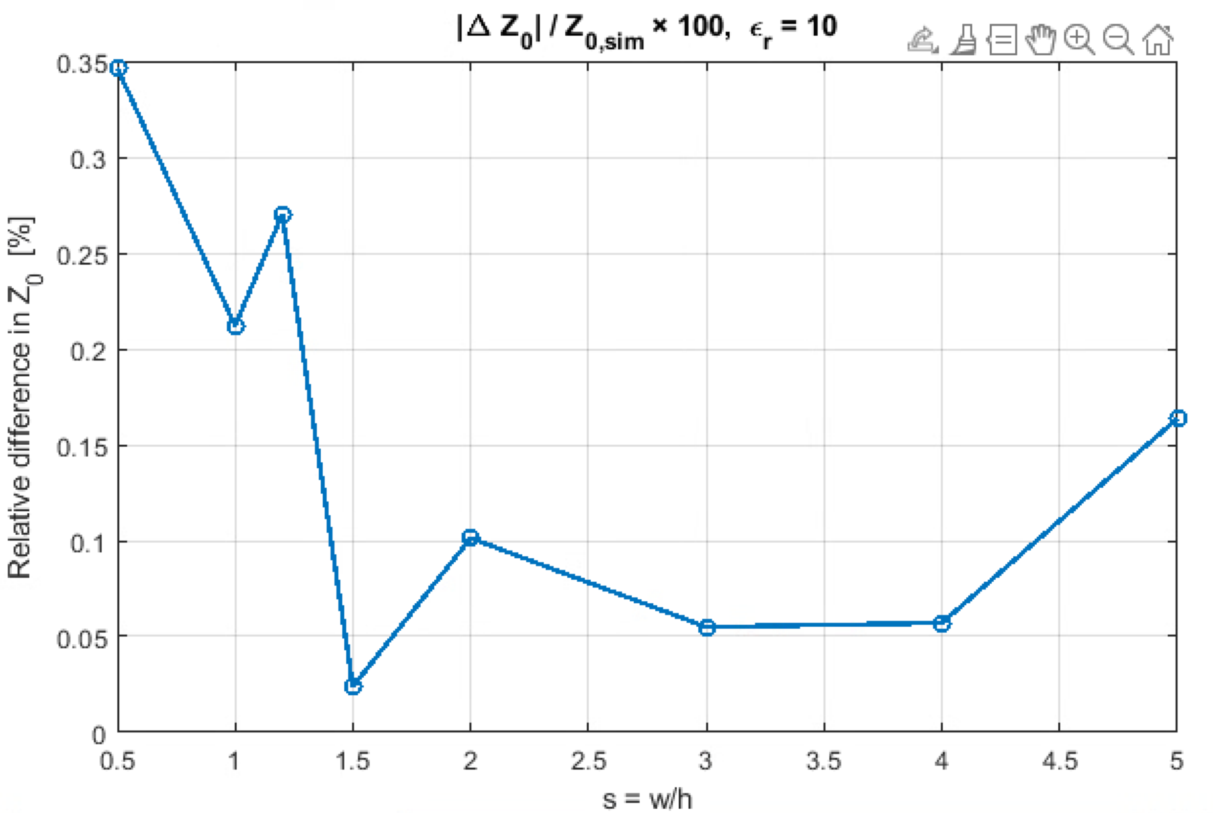

Figure 25: Relative difference (as a percentage) of Z_0 vs. s for Part 1

Part 2 – Relative difference of s vs. Z_0

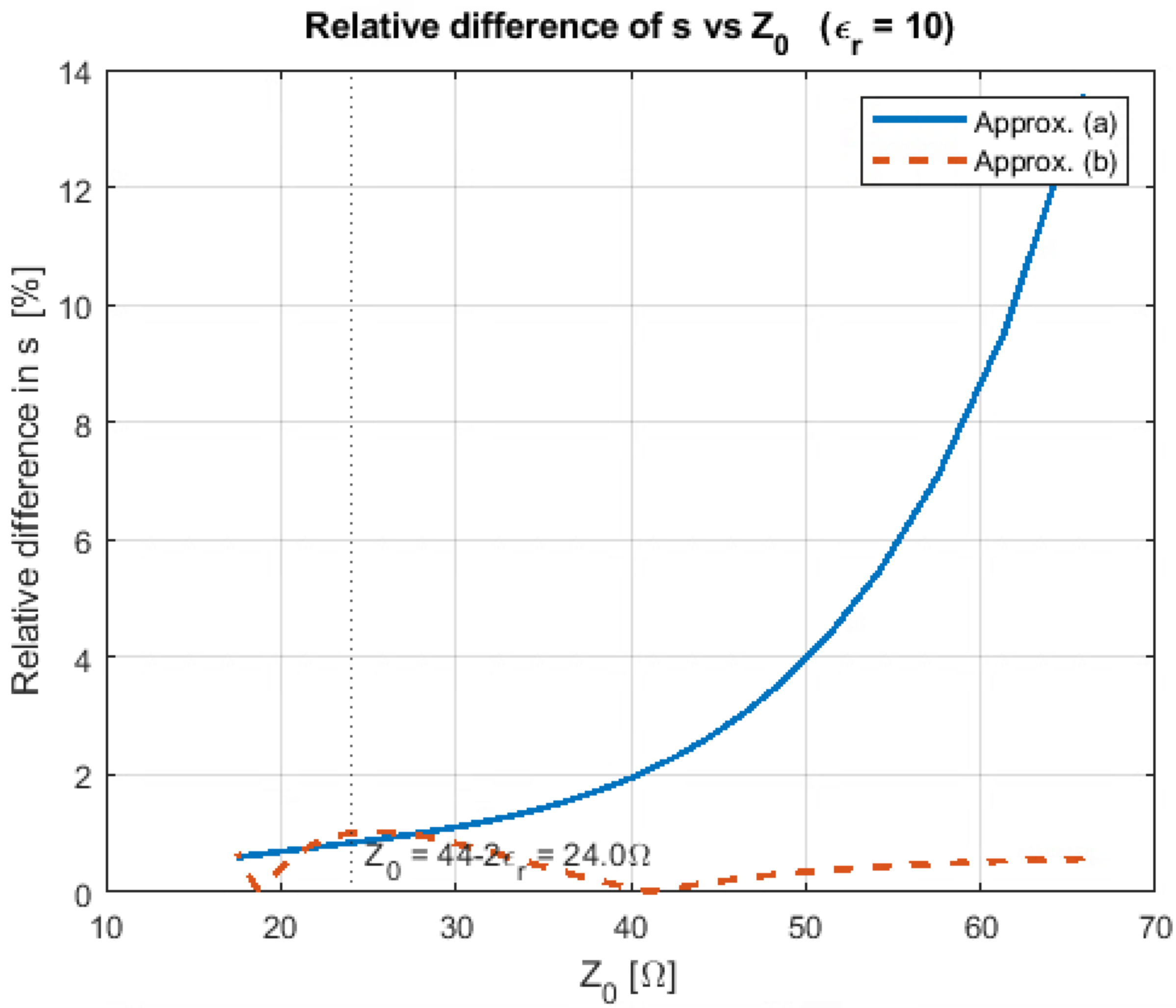

Figure 26: Relative difference (as a percentage) of s vs. Z_0 for $\varepsilon_r = 10$ Part 2

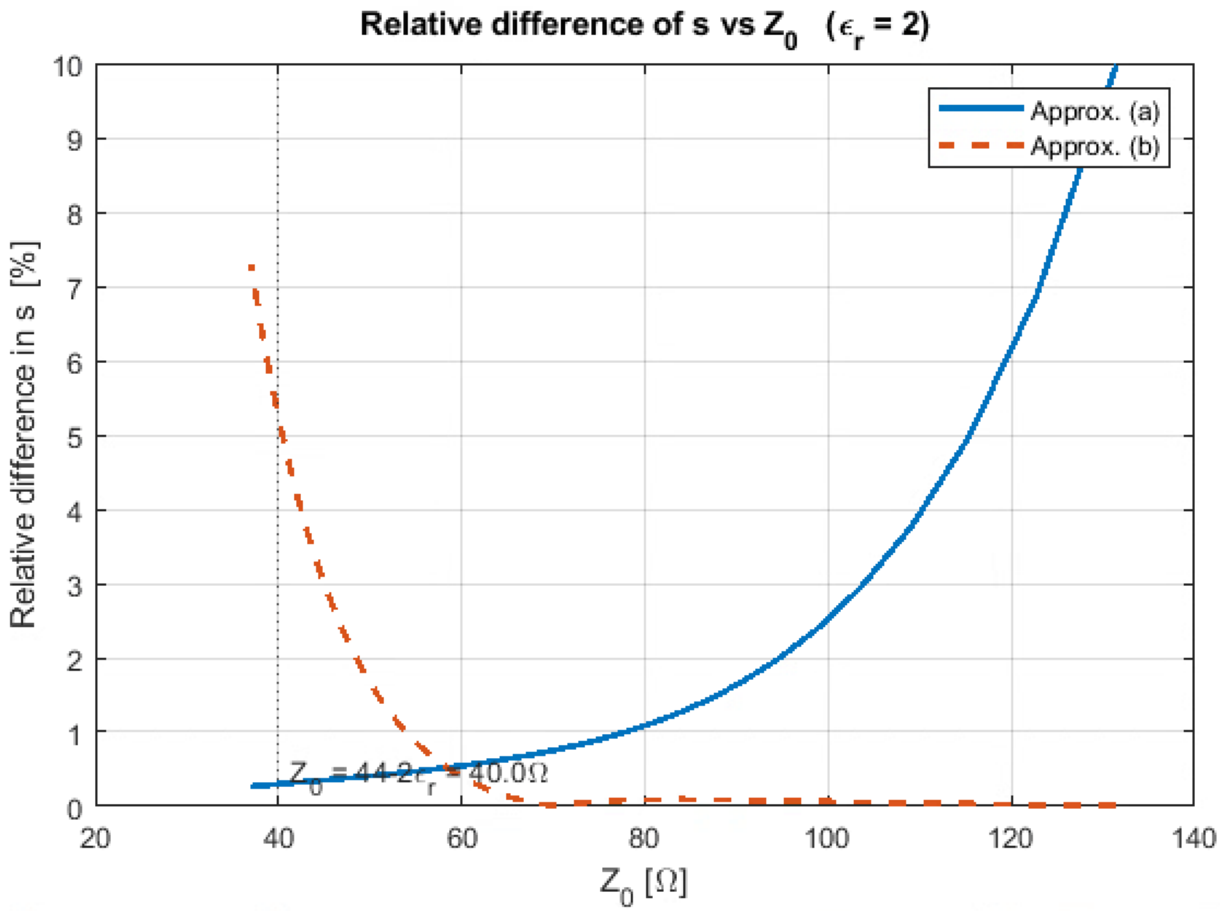

Figure 27: Relative difference (as a percentage) of s vs. Z_0 for $\varepsilon_r = 2$ Part 2

Discussion:

Part 1 – Relative difference of Z_0 vs. s

Across both substrates ($\varepsilon_r = 2$ and $\varepsilon_r = 10$), the characteristic impedance Z_0 decreases rapidly as the strip becomes wider relative to the substrate (s increases). Physically, a wider strip spreads the electric flux (denoted by the E and B field lines in *Figure 1.*) over a larger conductor area. This raises the capacitance per unit length **(**$C = \frac{\varepsilon A}{d}$**)** quicker than the inductance rises, so $Z_0 = \sqrt{L/C}$ falls. In our experiment, we kept ε_r fixed and varied s, we can make the important observation that for any fixed s, lowering ε_r yields a larger Z_0 because a lower permittivity reduces the effective dielectric loading (smaller C), therefore increasing Z_0. These qualitative trends are exactly what the forward model predicts which can was demonstrated in *Figure 24.*

Observing the relative difference of Z_0 in comparison of that of the Java application simulation, we notice several key tendencies. Over the mid-range of s, the relative difference (as a percentage) remains small and smooth ($\approx 1.5 - 4$). The plot grows near the extremes of the selected s values. A physical description of this observation is an increase in relative difference for very narrow strips ($s < 1.5$) and very wide strips ($s > 4$). This behavior is expected:

- The expressions used to calculate Z_0 are simply curve-fits to rigorous solutions. These approximations remain relatively stable and consistent for "sound" geometries (sound implying normally occurring). These errors grow as the geometry of the micro-strip approaches extreme conditions (very small or very large s).
- As with any software, small implementation differences exist between the given formulas (my code) and the applications simulations (e.g. internal constants and rounding) which show up primarily at both boundaries.

The Effect of Frequency:

Although the lossless, quasi-TEM characteristic impedance formulas we've been given are frequency-independent, our simulations still required a frequency choice to ensure the quasi-TEM assumptions held and to match the Java application inputs. In the *Appendix*, an equation has been derived to relate frequency to s. The final expression for quasi-TEM lines is $\lambda = \frac{c}{f\sqrt{\varepsilon_{\text{eff}}}}$ which we used to validate the criteria of $h/\lambda \ll 1$. By setting $f = 1$ GHz and $h = 1$ mm, I obtained $h/\lambda \approx 0.004$ for $\varepsilon_r = 2$ and ≈ 0.008 for $\varepsilon_r = 10$. These values are both comfortably below 0.01. Thus we chose the frequency to be 1GHz. It is important to note that other frequencies may be valid as long as they fulfill the given constraint. An additional observation was that even with the variation of frequency between 0Hz-20GHz, minimal differences in our characteristic impedance were realized. These differences were enough to significantly change our relative difference for Part 1.

Part 2 – Relative difference of s vs. Z_0

The assignment provides two approximations for $s = w/h$, each with a constraint based on Z_0(accuracy asserted too within $\approx$ 2%):

- **(a)** Use when $Z_0 \leq (44 - 2\varepsilon_r)\ \Omega$
- **(b)** Use when $Z_0 \geq (44 - 2\varepsilon_r)\ \Omega$

From these constraints the thresholds are:

- $\varepsilon_r = 10$: $Z_0 = 44 - 2 \cdot 10 = 24\ \Omega$
- $\varepsilon_r = 2$: $Z_0 = 44 - 2 \cdot 2 = 40\ \Omega$

The relative difference of s vs. Z_0 plots show the expected pattern:

- For $\varepsilon_r = 10$, approximation (a) performs best (≤2%) in the low-impedance region (below $\approx 24\ \Omega$). This can physically be described by a wider strip (bigger path for electron flow). Approximation (b) is most accurate in the high-impedance region (above $\approx 24\ \Omega$). This can physically be described by a thinner strip (smaller path for electron flow). A small bump in error typically appears near the **transition** because the two curve-fits were trained on different regimes.
- For $\varepsilon_r = 2$, the same story holds with the boundary near 40 Ω: (b) is the right tool for the higher impedance side (narrower strips), while (a) is appropriate for lower Z_0 (wider strips).

In short, the obtained plots confirm the assignments required 2% threshold for specific ranges of the characteristic impedance. The largest errors concentrated around the crossover between approximations where neither fit is dominant.

Explanation of the Reverse Engineered Formulas:

The approximation is highly nonlinear. Using two separate curve-fits allows each formula to specialize: (b) captures the logarithmic (fast) variation of s in the high-impedance (lower s) region, while (a) captures the slow-varying behavior in the low-impedance (higher s) region. These 2 regions keep each approximation suitable in its intended region. Around $Z_0 = (44 - 2\varepsilon_r)\ \Omega$, neither approximation is “dominant,” so the largest error peaks around this cross-over. Using the correct approximation (a vs. b) and staying a safe margin away from the boundary minimizes this error.

Conclusion:

Part 1 – Relative difference of Z_0 vs. s

In the first part of the study, I computed the characteristic impedance Z_0 as a function of the width-to-thickness ratio $s = w/h$ for $\varepsilon_r = 2$ and $\varepsilon_r = 10$ using the assignment's model. These values were then compared to those within the simulation. The results demonstrate a steady decrease of Z_0 with increasing s, and for any fixed geometry the lower-permittivity substrate produces a higher Z_0 than the higher-permittivity substrate. Over the range of selected s values, the compared results agreed closely within a percent difference of 0.35%. We observed a smooth relative difference within the vast majority of the plot and where the largest discrepancies appear only at the extremes of s. The frequency chosen for the virtual simulations (as justified in the *Appendix* by verifying $h/\lambda \ll 1$) simply ensured the quasi-TEM assumptions were valid. In conclusion, these observations confirm that the forward model captures the essential dependence of Z_0 on geometry and dielectric loading and that the simulation comparisons are consistent with theoretical expectations.

Part 2 – Relative difference of s vs. Z_0

In the second part, I inverted the design problem by solving for s from Z_0 and ε_r) using the two provided approximations, 4 curves where plot to find $Z_0 \approx 2\%$ accuracy. The data supports the assignment's constraint split: for $\varepsilon_r = 10$, approximation (a) performs as intended on the low-impedance side (below about 24 Ω), while approximation (b) is accurate on the high-impedance side (above that threshold). For $\varepsilon_r = 2$, the same pattern holds with the boundary near 40 Ω. The only persistent pockets of larger error occur in a small region around each crossover, which is expected because the curve-fits were trained for different regions. Overall, the inverse approximations reliably resulted in the target width ratios when applied in their proper domains, and the methodology successfully demonstrates where each should be used in practice.

References

Libretexts. (2022, May 22). *4.4: Microstrip transmission lines*. Engineering LibreTexts. https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Electronics/Book%3A_Fundamentals_of_Microwave_and_RF_Design_%28Steer%29/04%3A_Planar_Transmission_Lines/4.04%3A_Microstrip_Transmission_Lines

Appendix:

Deriving our frequency as a function of s:
General definitions (derived in class):

$$\gamma = j\beta$$
$$\beta = \omega\sqrt{\mu\epsilon}$$

Wavelength is:

$$\lambda = \frac{2\pi}{\beta} = \frac{1}{f\sqrt{\mu\varepsilon}}$$

We know that $\mu = \mu_0\mu_r$, $\varepsilon = \varepsilon_0\varepsilon_r$ and $c = \frac{1}{\sqrt{\mu_0\varepsilon_0}}$

$$\lambda = \frac{c}{f\sqrt{\mu_r\varepsilon_r}}$$

For non-magnetic dielectrics $\mu_r \approx 1$

$$\lambda = \frac{c}{f\sqrt{\varepsilon_r}}$$

For a guided quasi-TEM lines (microstrip), ε_r can be replaced with $\sqrt{\varepsilon_{eff}}$ (Libretexts, 2022).

$$\lambda = \frac{c}{f\sqrt{\varepsilon_{eff}}}$$

The quasi-TEM approximation depends on $\frac{h}{\lambda} \ll 1$. We will aim to drive this ratio to be less than 0.01. We can assume a frequency of 1GHz and test.
For a substrate thickness of $h = 1mm$:

- For $\varepsilon_r = 2$:

$$\lambda \approx \frac{3 * 10^8}{1 * 10^9 * \sqrt{1.5}} \approx 245mm$$
$$\frac{h}{\lambda} \approx 0.004$$

- For $\varepsilon_r = 10$:

$$\lambda \approx \frac{3 * 10^8}{1 * 10^9 * \sqrt{5.5}} \approx 128mm$$
$$\frac{h}{\lambda} \approx 0.008$$

These ratios are both in agreeance to $\frac{h}{\lambda} \ll 1$.