

Thrust Normal to Velocity Vector

July 15, 2021

Expressing $w(r)$ in terms of ϕ and β

Approach: reexpress components of $w(r)$ and substitute $r(\phi)$

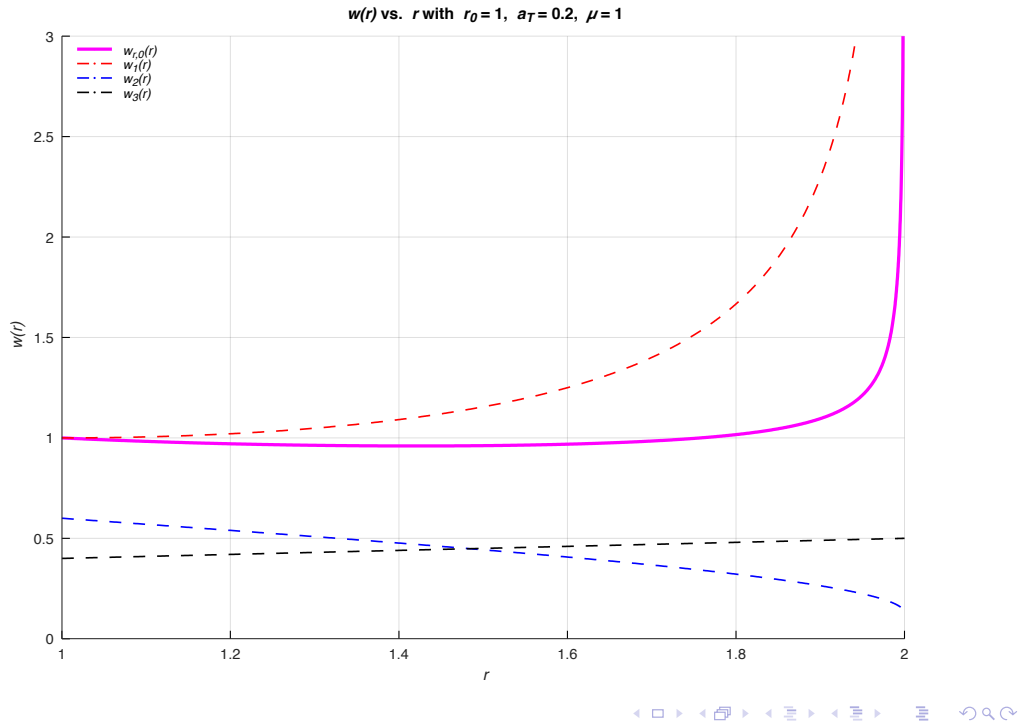
$$w(r) = \underbrace{\frac{r_0}{\sqrt{r(2r_0 - r)}}}_{w_1(r)} \underbrace{\left[1 - \frac{a_T r_0^2}{\mu} \left(3 \sin^{-1} \left(\sqrt{\frac{r}{2r_0}} \right) - \frac{3\pi}{4} + 2 \right) \right]}_{w_2(r)} + \underbrace{\frac{a_T r_0}{2\mu} (r + 3r_0)}_{w_3(r)}$$

For plots: $r_0 = 1, \mu = 1, a_T = 0.2$

Definitions: $k(\phi) = \phi + \pi/4$, $r(\phi) = 2r_0 \sin^2(k)$, $\beta = \sqrt{\frac{4\mu}{(3\pi+8)a_T r_0^2}}$

Subscript p denotes function of ϕ .

Plotting $w_1(r)$, $w_2(r)$, $w_3(r)$



Reexpressing $w_1(r)$

$$w_{1p}(\phi) = \frac{r_0}{\sqrt{r(2r_0 - r)}} \Big|_{r(\phi)} \quad (1)$$

$$= \frac{r_0}{\sqrt{2r_0 \sin^2(k)(2r_0 - 2r_0 \sin^2(k))}} \quad (2)$$

$$= \frac{r_0}{\sqrt{4r_0^2 \sin^2(k) \cos^2(k)}} \quad (3)$$

$$= \frac{1}{\sin(2k)} \quad (4)$$

Obtain 3 using $\cos^2(k) = 1 - \sin^2(k)$ and 4 by $\sin(2k) = 2 \sin(k) \cos(k)$.

Reexpressing $w_2(r)$

$$w_{2p}(\phi) = 1 - \frac{a_T r_0^2}{\mu} \left(3 \arcsin\left(\sqrt{\frac{r}{2r_0}}\right) - \frac{3\pi}{4} + 2 \right) \Big|_{r(\phi)} \quad (5)$$

$$= 1 - \frac{a_T r_0^2}{\mu} \underbrace{\left(3k - \frac{3\pi}{4} + 2 \right)}_{3\phi+2} \quad (6)$$

$$= 1 - \frac{a_T r_0^2}{\mu} (3\phi + 2) \quad (7)$$

[illegible]Reexpressing $w_3(r)$

$$w_{3p}(\phi) = \frac{a_T r_0}{2\mu} (r + 3r_0) \Big|_{r(\phi)} \quad (8)$$

$$= \frac{a_T r_0}{2\mu} (r_0(2 \sin^2(k) + 3)) \quad (9)$$

$$= \frac{a_T r_0^2}{2\mu} (2 \sin^2(k) + 3) \quad (10)$$

$$= \frac{2}{\beta^2(3\pi + 8)}(2\sin^2(k) + 3) \quad (11)$$

Obtain 11 by $\beta = \sqrt{\frac{4\mu}{(3\pi+8)a_T r_0^2}} \Rightarrow \frac{1}{\beta^2} = \frac{(3\pi+8)a_T r_0^2}{4\mu}$

[illegible]

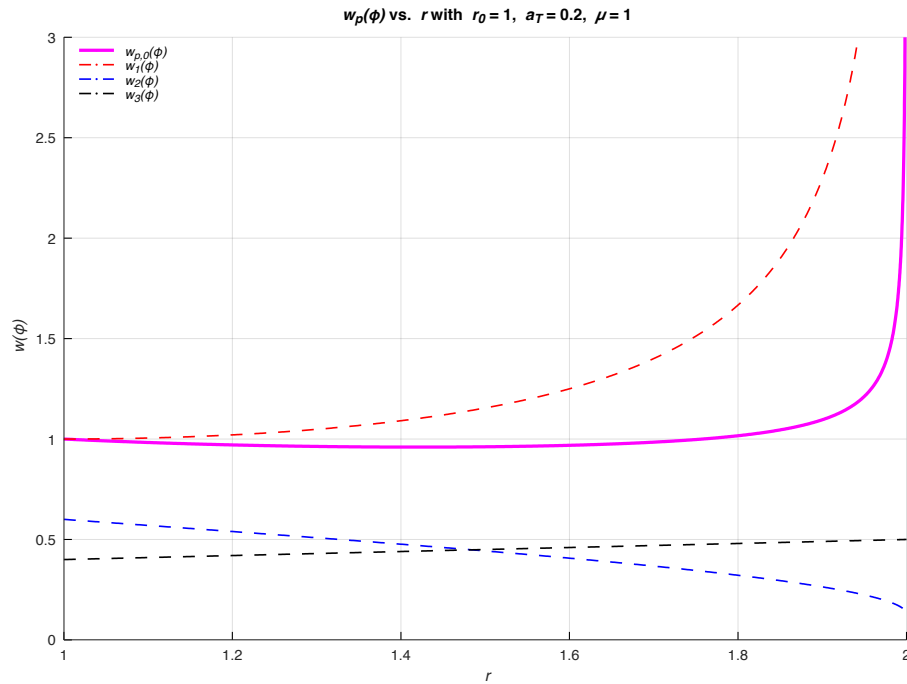
Expression for $w_p(\phi)$

$$w_p(\phi) = \frac{1}{\sin(2k)} \left[1 - \frac{a_T r_0^2}{\mu} (3\phi + 2) \right] + \frac{2}{\beta^2(3\pi + 8)} (2\sin^2(k) + 3) \quad (12)$$

$$= w_{1p}(\phi) w_{2p}(\phi) + w_{3p}(\phi) \quad (13)$$

Navigation icons: back, forward, search, etc.

Checking answer with plots



Navigation icons: back, forward, search, etc.

Expressing $y(r)$ in terms of ϕ and β

Approach similar to $w(r)$:

$$y(r) = \underbrace{\sqrt{\frac{\mu(2r_0 - r)}{rr_0}}}_{y_1(r)} \underbrace{\sqrt{1 - w(r)^2}}_{y_2(r)}$$

A set of navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

Reexpressing $y_1(r)$

$$y_{1p}(\phi) = \sqrt{\frac{\mu(2r_0 - r)}{rr_0}} \Big|_{r(\phi)} \quad (14)$$

$$= \sqrt{\frac{\mu 2r_0(1 - \cos^2(k))}{2r_0^2 \sin^2(k)}} \quad (15)$$

$$= \sqrt{\frac{\mu 2r_0 \cos^2(k)}{2r_0^2 \sin^2(k)}} \quad (16)$$

$$= \sqrt{\frac{\mu}{r_0}} \cot(k) \quad (17)$$

[illegible]

Reexpressing $y_2(r)$

Given $y_{2p}(\phi) = \sqrt{1 - w_p(\phi)^2}$.

Define $\varphi(\phi)$ such that $\frac{\varphi(\phi)}{(3\pi+8)^2\beta^4\sin^2(2k)} = 1 - w_p(\phi)^2$.

Then, by squaring 12,

$$1 - w_p(\phi)^2 = \frac{(3\pi+8)^2\beta^4\sin^2(2k) - [(3\pi+8)^2\beta^4 - 8(3\phi+2)(3\pi+8)\beta^2 + 16(3\phi+2)^2]}{(3\pi+8)^2\beta^4\sin^2(2k)} \quad (18)$$

$$- \frac{2\sin(2k)(4\sin^2(k) + 6)((3\pi+8)\beta^2 - 4(3\phi+2))}{(3\pi+8)^2\beta^4\sin^2(2k)} \quad (19)$$

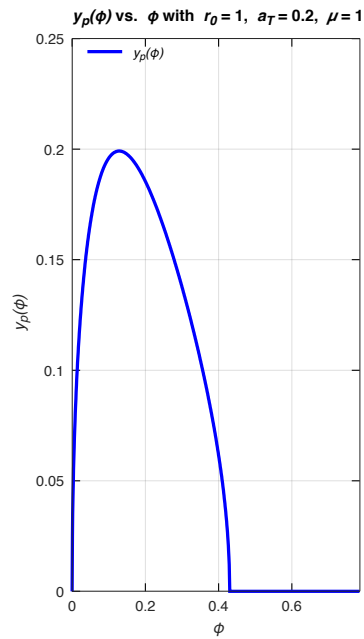
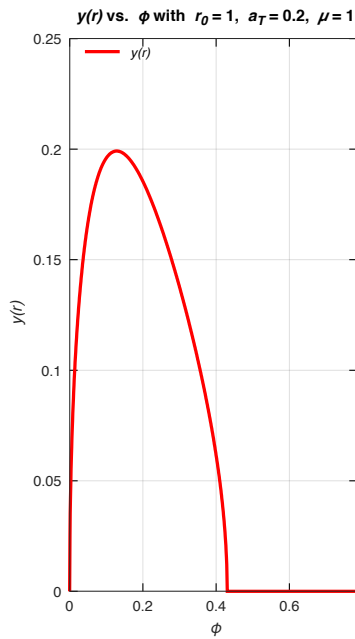
$$- \frac{\sin^2(2k)(16\sin^4(k) + 48\sin^2(k) + 36)}{(3\pi+8)^2\beta^4\sin^2(2k)} \quad (20)$$

Extracting the denominator,

$$y_p(\phi) = \frac{\sqrt{\varphi(\phi)}}{\beta^2(3\pi+8)\sin(2k)} \sqrt{\frac{\mu}{r_0}} \cot(k) \quad (21)$$

Navigation icons: back, forward, search, etc.

Plots to verify $y_p(\phi)$



Navigation icons: back, forward, search, etc.

Minimizing $w(r)$ or $w_p(\phi)$

- ▶ Began by differentiating $w(r)$ and simplifying
- ▶ Substituted $r(\phi)$ to find $w_p'(\phi)$
- ▶ Solving for ϕ in $w_p'(\phi)$ would not be feasible by hand since there were trigonometric and linear terms of ϕ
 - ▶ Obtained a greatly simplified expression that was used for minimization
- ▶ Used Octave to minimize the expression numerically

◀ ◻ ▶ ◀ [stack] ▶ ◀ [list] ▶ ◀ [list] ▶ [list] ↺ 🔍 ↻

First derivative test on $w(r)$

$$w_1'(r) = \frac{d}{dr} r_0(2r_0r - r^2)^{-1/2} \quad (22)$$

$$= -\frac{1}{2}r_0(2r_0 - 2r)(2r_0r - r^2)^{-3/2} \quad (23)$$

$$= \frac{-r_0(r_0 - r)}{(2r_0r - r^2)^{3/2}} \quad (24)$$

$$w_2'(r) = \frac{-3a_T r_0^2}{\mu} \frac{d}{dr} \arcsin \sqrt{\frac{r}{2r_0}} \quad (25)$$

$$= \frac{-3a_T r_0^2}{\mu} \frac{\frac{1}{4r_0} \left(\frac{r}{2r_0}\right)^{-\frac{1}{2}}}{\sqrt{1 - \frac{r}{2r_0}}} \quad (26)$$

$$= \frac{-3a_T r_0}{\mu} \frac{1}{4 \sqrt{\frac{r}{2r_0} \left(\frac{2r_0 - r}{2r_0} \right)}} \quad (27)$$

$$= -\frac{3a_T r_0}{2\mu\sqrt{2r_0 r - r^2}} \quad (28)$$

$$w_3'(r) = \frac{d}{dr} \frac{a_T r_0}{2\mu} (r + 3r_0) \quad (29)$$

$$= \frac{a_T r_0}{2\mu} \quad (30)$$

A set of navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

First derivative test on $w(r)$ (continued)

$$w'(r) = \frac{d}{dr} [w_1(r)w_2(r) + w_3(r)] \quad (31)$$

$$= w_1'(r)w_2(r) + w_1(r)w_2'(r) + w_3'(r) \quad (32)$$

Substituting expressions found prior,

$$w'(r) = \underbrace{\frac{-r_0(r_0 - r)}{(2r_0r - r^2)^{3/2}}}_{w_1'(r)} \underbrace{\left[1 - \frac{a_T r_0^2}{\mu} \left(3 \arcsin\left(\sqrt{\frac{r}{2r_0}}\right) - \frac{3\pi}{4} + 2 \right) \right]}_{w_2(r)} \quad (33)$$

$$\underbrace{-\frac{3a_T r_0}{2\mu(2r_0 r - r^2)}}_{w_1(r)w_2'(r)} + \underbrace{\frac{a_T r_0}{2\mu}}_{w_3(r)} \quad (34)$$

A set of navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

Reubstituting with $r(\phi)$ and simplifying

First, $2r_0r - r^2 = 4r_0^2 \sin^2(k) - 4r_0 \sin^4(k) = 4r_0^2 \sin^2(k) \cos^2(k)$.

$$w_{1p}'(\phi) = \frac{2r_0^2 \sin^2(k) - r_0^2}{(2r_0 \sin(k) \cos(k))^3} = \frac{2 \sin^2(k) - 1}{r_0 \sin^3(2k)} \quad (35)$$

$$w_{2p}(\phi) = 1 - \frac{a_T r_0^2}{\mu} (3\phi + 2) \quad (36)$$

$$w_1(r)w_2'(r) = -\frac{3a_T r_0}{2\mu(4r_0^2 \sin^2(k) \cos^2(k))} = -\frac{3a_T r_0}{2\mu \sin^2(2k)} \quad (37)$$

Then,

$$w_p'(\phi) = \frac{2\sin^2(k) - 1}{r_0\sin^3(2k)} \left(1 - \frac{a\tau r_0^2}{\mu}(3\phi + 2) \right) - \frac{3a\tau r_0}{2\mu\sin^2(2k)} + \frac{a\tau r_0}{2\mu} \quad (38)$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ≡ ≡ ↺ 🔍 ↻

Reubstituting with $r(\phi)$ and simplifying (continued)

$$w_p'(\phi) = \underbrace{\frac{2 \sin^2(k) - 1}{r_0 \sin^3(2k)} \left(1 - \frac{a_T r_0^2}{\mu} (3\phi + 2) \right)}_{\frac{-2\mu \cos(2k) \left(1 - \frac{a_T r_0^2}{\mu} (3\phi + 2) \right)}{2\mu r_0 \sin^3(2k)}} - \underbrace{\frac{3a_T r_0}{2\mu \sin^2(2k)} + \frac{a_T r_0}{2\mu}}_{\frac{a_T r_0^2 \sin^3(2k) - 3a_T r_0^2 \sin(2k)}{2\mu r_0 \sin^3(2k)}} \quad (39)$$

$$= \frac{a_T r_0^2 (\sin^2(2k) - 3) \sin(2k) - 2\mu \cos(2k) \left(1 - \frac{a_T r_0^2}{\mu} (3\phi + 2)\right)}{2\mu r_0 \sin^3(2k)} \quad (40)$$

A set of navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

Minimization of $w_p(\phi)$

Note that $2\mu r_0 \sin^3(2k) \neq 0 \implies \sin(2k) \neq 0$. So $2(\phi + \pi/4) \notin \{0, \pi\}$ means $\phi \neq \pi/4$ since $\phi \in [0, \pi/4)$.

$$w_p'(\phi) = \frac{a_T r_0^2 (\sin^2(2k) - 3) \sin(2k) - 2\mu \cos(2k) \left(1 - \frac{a_T r_0^2}{\mu} (3\phi + 2)\right)}{2\mu r_0 \sin^3(2k)} = 0 \quad (41)$$

$$\Rightarrow a_T r_0^2 (\sin^2(2k) - 3) \sin(2k) - 2\mu \cos(2k) \left(1 - \frac{a_T r_0^2}{\mu} (3\phi + 2)\right) = 0 \quad (42)$$

$$\frac{\sin(2k)}{2\mu \cos(2k)} = \underbrace{\frac{\tan(2k)}{2\mu}}_{d_1(\phi)} = \frac{1 - \frac{a_T r_0^2}{\mu}(3\phi + 2)}{\underbrace{a_T r_0^2(\sin^3(2k) - 3)}_{d_2(\phi)}} \quad (43)$$

Numerically finding the intersection of $d_1(\phi)$ and $d_2(\phi)$ yields the value of ϕ at $\min(w_p(\phi))$, which can then be used to find r for $\min(w(r))$.

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ≡ ≡ ↺ 🔍 ↻

Numerical minimization of $w_p(\phi)$

Code:

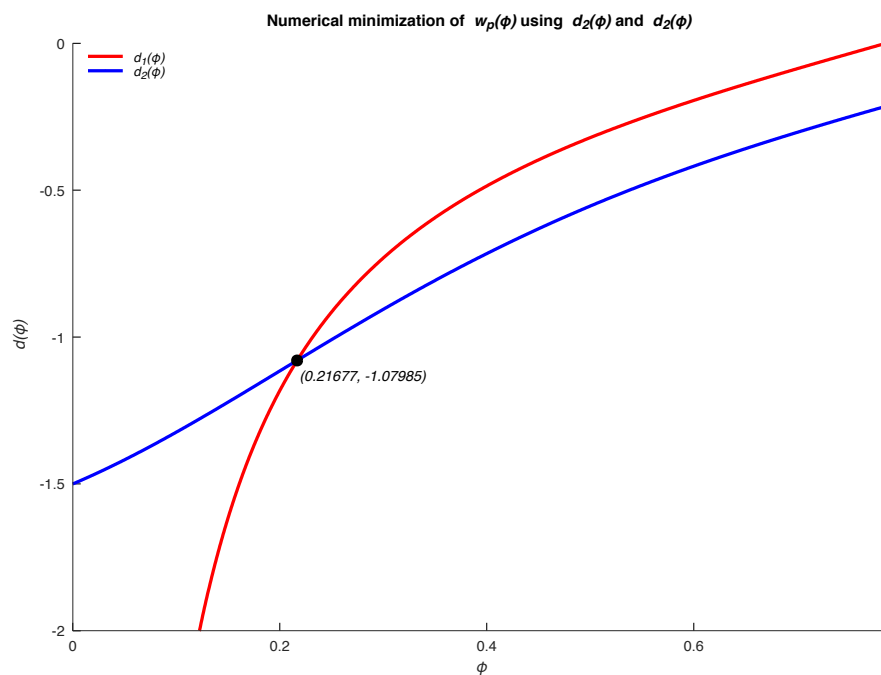
```
d1 = tan(2*k)/(2*mu);  
d2 = (1-a_T*r_0^2/mu * (3*phi+2))./(a_T*r_0^2*(sin(2*k).^2-3));  
  
% minimization  
intersect = find(abs(d1 - d2) <= min(abs(d1 - d2)));  
  
ix = phi(intersect)  
iy = mean([d1(intersect) d2(intersect)])  
r_min = r(intersect)
```

Output:

```
ix = 0.2168  
iy = -1.0799  
r_min = 1.4201
```

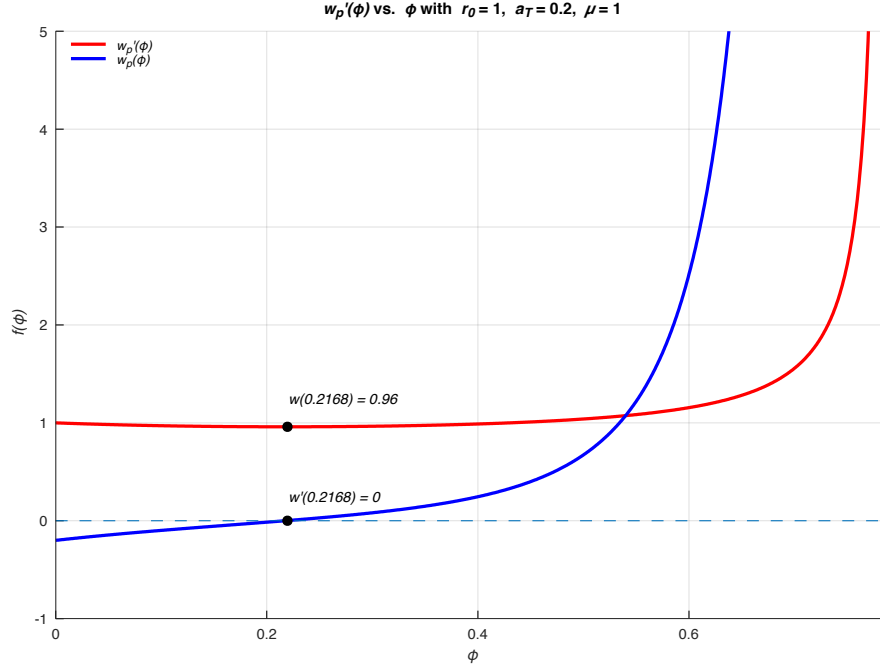
Navigation icons: back, forward, search, etc.

Intersection of $d_1(\phi)$ and $d_2(\phi)$



Navigation icons: back, forward, search, etc.

Visual of $\min(w_p(\phi))$



1 Simplifying expression for $w_p(\phi)$

The sine function is a phase shift of the cosine function, so

$$\sin(2k) = \sin(2\phi + \pi/2) = \cos(2\phi). \quad (1)$$

It is also given that

$$2 \sin^2(2k) = 1 - \cos(2k). \quad (2)$$

Because $\cos(2k) = -\sin(2\phi)$ by a similar argument,

$$2 \sin^2(2k) = 1 + \sin(2\phi). \quad (3)$$

Finally

$$w_p(\phi) = \frac{1}{\underbrace{\sin(2\phi + \pi/2)}_{\cos(2\phi)}} \left[1 - \underbrace{\frac{a_T r_0^2}{\mu}}_{-\frac{4}{\beta^2(3\pi+8)}} (3\phi + 2) \right] + \frac{2}{\beta^2(3\pi+8)} \underbrace{(2 \sin^2(\phi + \pi/4) + 3)}_{\sin(2\phi)+4} \quad (4)$$

$$= \frac{1}{\cos(2\phi)} \left[1 - \frac{4}{\beta^2(3\pi+8)} (3\phi + 2) \right] + \frac{2}{\beta^2(3\pi+8)} (\sin(2\phi) + 4) \quad (5)$$

2 Numerical evaluation of $y(\phi_0) = 0$

Notation: Let ϕ_0 satisfy $y(\phi_0) = 0$, $w_p(\phi_0) = 1$ and ϕ_{\min} minimize $w_p(\phi)$.

To evaluate $y_p(\phi) = 0$, we first define `phi = (1:1:1000)*pi/4/1000`. Using this vector to formulate `y_p(Beta)`, where `Beta` is a variable constant, we have a vector of length 1000 representing $y_p(\phi)$. Note that

$$y_p(\phi) = \frac{\sqrt{\varphi(\phi)}}{\beta^2(3\pi + 8)\sin(2\phi + \pi/2)} \sqrt{\frac{\mu}{r_0}} \cot(\phi + \pi/4) \quad (6)$$

$\varphi(\phi)$ was defined in Eq. 18-20 of the presentation, and the above expression comes from Eq. 21 of the presentation. This implies that $\sin(2\phi + \pi/2) \neq 0$, and further that y_p is only defined for $\varphi(\phi) \geq 0$. Thus, the vector `y_p(Beta)` must be shortened to reflect this and exclude complex numbers. From calculations, we find that the real representation of y_p is `y_p(Beta)(1:548)` using $a_T = 0.2, r_0 = 1, \mu = 1$ with $\beta \approx 1.07$. To find the section of this vector that is $\in \mathbb{R}$, we have called a function `realBreakpoint(vector)` in Figure 1. Sorting this vector and determining the corresponding ϕ_0 solves the problem, where we expect `y_minValue = 0` and `phi_0` to be the corresponding value of ϕ :

```
[y_values index_vector] = sort(y_p(Beta)(1:548));
y_minValue = y_values(1);
phi_0 = index_vector(1);
```

This approach can be extended for a changing parameter β . Since $\delta = \frac{1}{\beta^2} \in (0.001, 1)$, $\beta = \sqrt{\frac{1}{\delta}} \in (1, \sqrt{1000})$. In Octave/MATLAB, we can implement it as

```
delta = (1:1:1000)/1000;
Beta_vec = sqrt(1./delta);
```

Mathematically, we say that

$$\vec{\beta} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \text{ and } \vec{\delta} = \begin{bmatrix} 1/\beta_1^2 \\ \vdots \\ 1/\beta_n^2 \end{bmatrix} \quad (7)$$

where we choose $n = 1000$. Thus, by iterating through the values of `Beta_vec`, we can generate a corresponding vector `phi_m1` to represent how ϕ_0 changes with respect to δ :

```

phi_m1 = zeros(1,n);
w_1 = zeros(1,n);
w_min = zeros(1,n);
y_0 = zeros(1,n);

for i = 1:n
    y_real = y_p(Beta_vec(i))(1:realBreakpoint(y_p(Beta_vec(i))));
    [yvals idx] = sort(y_real);
    if (length(idx) > 1)
        w_1(i) = w_p(Beta_vec(i))(idx(2));
        phi_m1(i) = phi(idx(2));
        y_0(i) = yvals(2);
    endif
end

```

Figure 1: Numerically finding $\vec{\phi}_0$ over $\vec{\delta}$

In Figure 1 above, the vector $\mathbf{y_0}$ is updated with the value $y_p(\phi_{0,i})$ in each iteration for each value of β , and is expected to be 0. The vector $\mathbf{w_1}$ is updated with $w_p(\phi_{m,i})$, which is expected to be unity. As was done with $\mathbf{y_p(Beta)}$, we have defined a vector $\mathbf{w_p(Beta)}$ to represent $w_p(\phi)$ where \mathbf{Beta} is a constant that can be varied. Thus, we should expect $w_p(\vec{\phi}_{\min}) = \vec{1}$, $y_p(\vec{\phi}_0) = \vec{0} \in \mathbb{R}^n$, and the

existence of $\vec{\phi}_m = \begin{bmatrix} \phi_{m,1} \\ \vdots \\ \phi_{m,n} \end{bmatrix} \in \mathbb{R}^n$ at the end of the loop. The second index of $\mathbf{y_p(Beta_vec(i))}$

is accessed to find $\phi_{0,i}$ within the loop because $\phi = 0$ always satisfies $y_p(\phi) = 0$, and we want to find the second such value. The conditional check is to account for cases where the domain of $y_p(\phi) \in \mathbb{R}$ is very small (i.e. $\text{length}(\mathbf{y_p(Beta_vec(i))})$ is 1), so we are only able to find $\phi_{0,i} = 0 \implies y(\phi_0) = 0$.

The results of ϕ_0 vs. δ are expressed in Figure 2 below. $y_p(\phi_0) \approx 0$ and $w_p(\phi_{\min}) \approx 1$ within numerical error, which verifies the validity of these results.

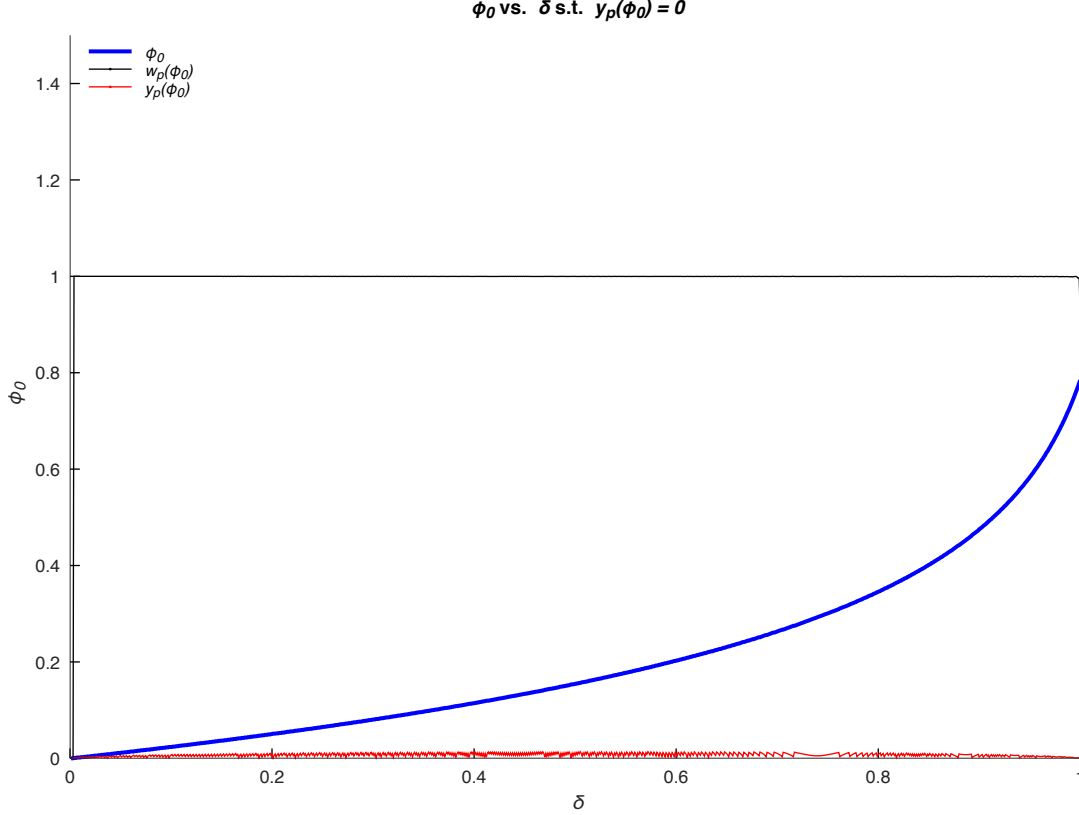


Figure 2: A plot of these results

3 Numerical minimization of $w_p(\phi)$

We use a similar approach as before to minimize $w_p(\phi)$ numerically. In Figure 1, `w_1(i)` was updated to reflect the value of $w_p(\phi_{\min,i})$ where $\phi = \phi_{\min,i}$ minimized $y_p(\phi)$ for corresponding values of β_i, δ_i from Eq. 7. The following code defines a vector `phi_m2` of length $n = 1000$ and populates it with the value of $\phi_{\min,i}$ that minimizes $w_p(\phi)$ for β_i . The corresponding minimum values are stored in another vector `w_min`. Note that the first index of the sorted vector `w_vals` is accessed, since we are looking for the absolute minimum.

```
phi_m2 = zeros(1,n);
w_min = zeros(1,n);
for i = 1:n
    [w_vals idx] = sort(w_p(Beta_vec(i)));
    phi_m2(i) = phi(idx(1));
    w_min(i) = w_vals(1);
end
```

Figure 3: Iteratively finding $\vec{\phi}_{\min}$ for various β

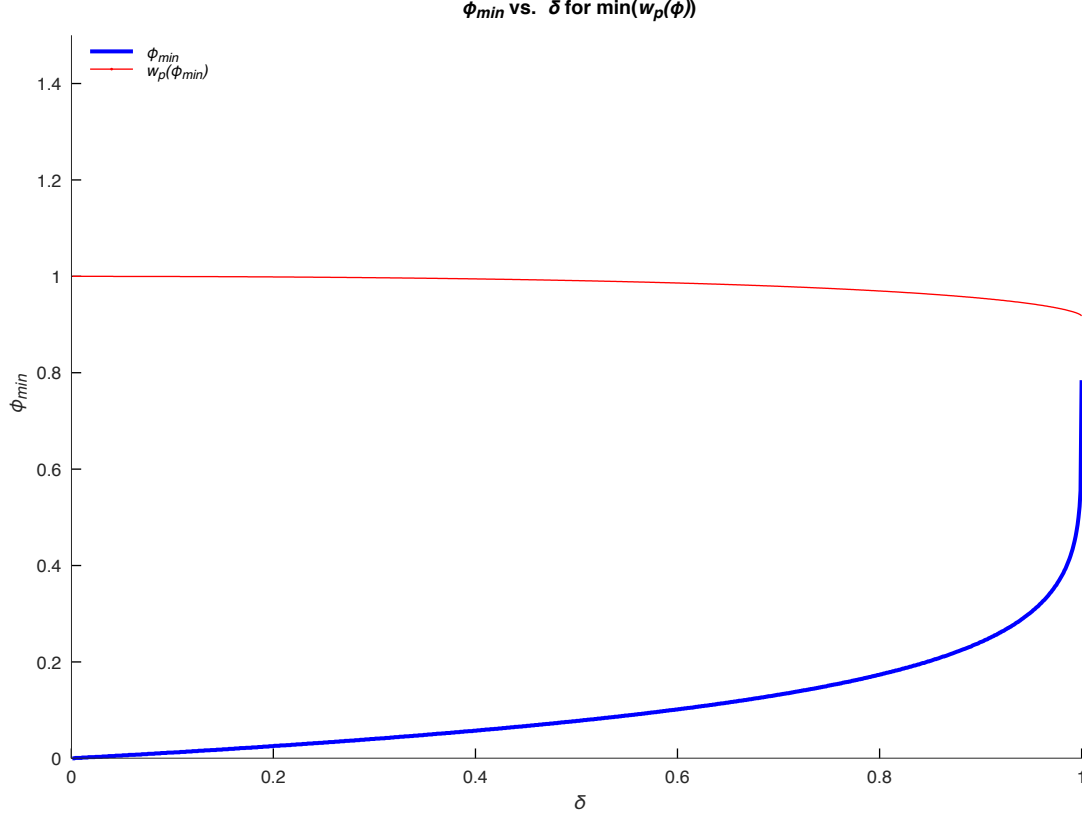


Figure 4: Values of ϕ_{\min} that minimize $w_p(\phi)$ for various β

Figure 4, showing ϕ_{\min} (representing phi_m2) vs. δ , summarizes these results.

4 Bisection and Newton-Raphson method for finding roots

4.1 Bisection method

We will use the bisection method to find when $y_p(\phi) = 0$ for a given tolerance ϵ . The requirement for the bisection method is that the function in question is continuous on the closed interval $[a, b]$ ¹, so the roots of $y_p(\phi)$ can be found. The roots of $w'(\phi)$ will be found using the Newton's method later.

4.2 Newton-Raphson method

In Eq. 40 from the presentation, $w_p'(\phi)$ was not expressed in terms of β , so the following expression will be used:

$$w_p'(\phi) = \frac{2(\sin^2(2\phi + \pi/2) - 3)\sin(2\phi + \pi/2)}{(3\pi + 8)\beta^2 \sin^3(2\phi + \pi/2)} - \frac{\cos(2\phi + \pi/2) \left(1 - \frac{4}{(3\pi + 8)\beta^2}(3\phi + 2)\right)}{r_0 \sin^3(2\phi + \pi/2)} \quad (8)$$

¹[Bisection method - Wolfram MathWorld](#)

This method cannot be used to solve for the root of $y_p(\phi)$ because although the function exists in the domain $\phi \in [0, \pi/4)$, $\lim_{\phi \rightarrow 0^+} y_p'(\phi)$ and $\lim_{\phi \rightarrow \pi/4^-} y_p'(\phi)$ do not exist, so the function is not differentiable on the entire interval².

From observing the graph of $y_p(\phi)$ on Page 6, this is confirmed visually. Since the endpoints of $y_p(\phi)$ on this interval are its two roots, they cannot be found through this method. The bisection method works, however, since the only requirement is continuity of $y_p(\phi)$ on the interval.

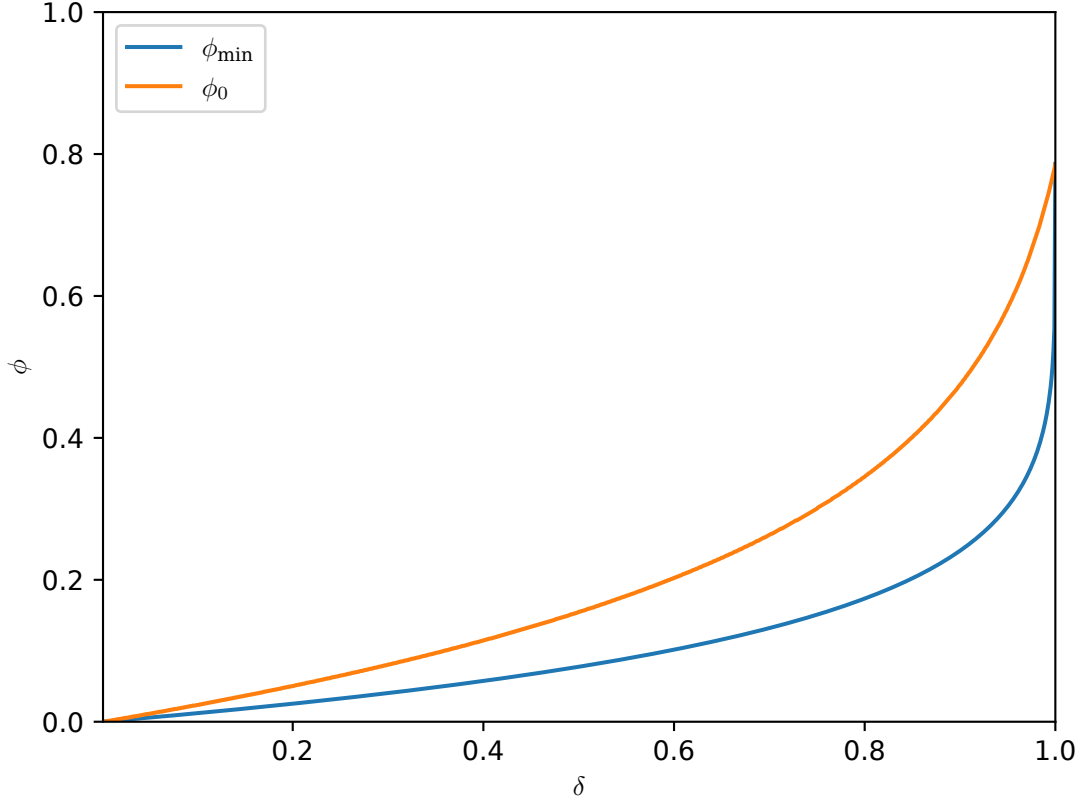


Figure 5: A plot of ϕ_{\min} and ϕ_0 vs δ

The code used to implement both algorithms is located [here](#). Figure 5 demonstrates the bisection method for finding ϕ_0 and Newton's method for ϕ_{\min} .

5 Curve fitting

5.1 Objective

The least-squared method of curve fitting³ was used to generate the closed-form equations $\phi_0(\delta)$ and $\phi_{\min}(\delta)$.

The approach is to define an error function $E(\phi) = \sqrt{\frac{1}{n} \sum_{i=1}^n (\phi_i - \phi(\delta_i))^2}$ that represents the RMS error between the fitted curve and original data. n is the number of data points in the set

²Newton's method

³Curve fitting: least squares methods

$\{(\delta_1, \phi_1), \dots, (\delta_n, \phi_n)\}$ and $\phi(\delta)$ represents the fitted curve. Because a polynomial curve is desired, the coefficients of the terms in $\delta(\phi)$ belong to $\vec{c} \in \mathbb{R}^{k+1}$, so there are $k+1$ coefficients in the fitted polynomial. Since $\phi(\delta; \vec{c})$, $E : \mathbb{R}^{k+1} \rightarrow \mathbb{R}; \vec{c}$. The objective is then to find a \vec{c} which minimizes $E(\vec{c})$. Similarly, each $\phi(\delta_i)$ becomes a function of \vec{c} .

Minimizing $E(\vec{c}) = \sqrt{\frac{1}{n} \sum_{i=1}^n (\phi_i - \phi(\delta_i))^2}$ is the same as minimizing $E_1(\vec{c}) = \sum_{i=1}^n (\phi_i - \phi(\delta_i))^2$, so the next step is to solve

$$\frac{\partial E_1}{\partial c_j} = 2 \sum_{i=1}^n (\phi_i - \phi(\delta_i)) \frac{\partial \phi(\delta_i)}{\partial c_j} = 0 \quad (9)$$

$$\sum_{i=1}^n \frac{\partial \phi(\delta_i)}{\partial c_j} \phi_i = \sum_{i=1}^n \frac{\partial \phi(\delta_i)}{\partial c_j} \phi(\delta_i) \quad (10)$$

$$\sum_{i=1}^n \delta_i^{j-1} \phi_i = \sum_{i=1}^n c_j \delta_i^{2j-1} + \dots + c_1 \delta_i^{j-1} = \sum_{\ell=1}^j \sum_{i=1}^n c_{j-\ell+1} \delta_i^{2j-\ell-1} \quad (11)$$

where $j = 1, \dots, k+1$ and the polynomial is of degree k .

A vectorized implementation yields

$$\begin{bmatrix} \sum_{i=1}^n \delta_i^k \phi_i \\ \vdots \\ \sum_{i=1}^n \delta_i^1 \phi_i \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n \delta_i^{2k} & \cdots & \sum_{i=1}^n \delta_i^k \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^n \delta_i^k & \cdots & \sum_{i=1}^n 1 \end{bmatrix} \begin{bmatrix} c_{k+1} \\ \vdots \\ c_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n c_{k+1} \delta_i^{2k} & \cdots & \sum_{i=1}^n c_1 \delta_i^k \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^n c_{k+1} \delta_i^k & \cdots & \sum_{i=1}^n c_1 \end{bmatrix} \quad (12)$$

$$\vec{b} = A\vec{c} \quad (13)$$

The matrix-vector equation $A\vec{c} = \vec{b}$ can then be solved for \vec{c} using an optimized linear algebra library, and the resulting coefficients used to generate a function $\phi(\delta)$.

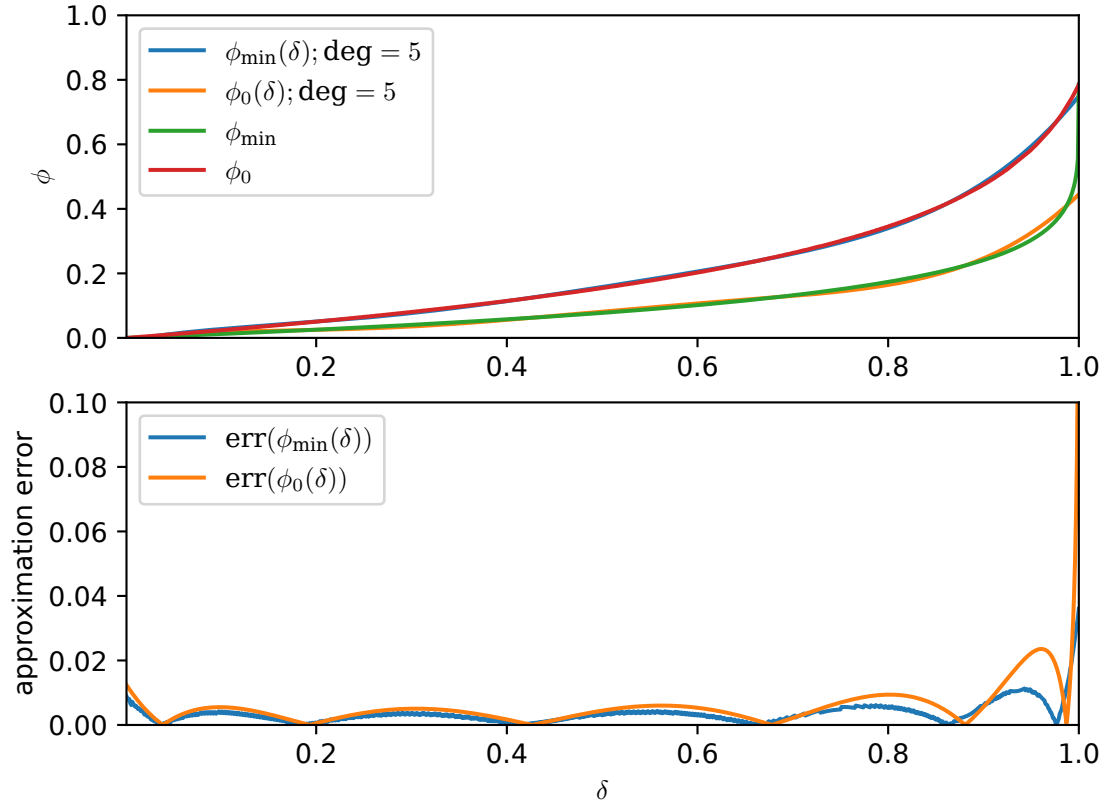


Figure 6: Using degree 5 polynomials to approximate $\phi_0(\delta)$ and $\phi_{\min}(\delta)$

A better approximation is desired since even with degree 5, a polynomial approximation cannot fit well to the smooth curves of ϕ_0 and ϕ_{\min} . This is evidenced by the Runge effect visible in the error plot of Figure 6.

5.2 Attempted approximation using Chebyshev polynomial interpolation

The Chebyshev polynomial approximations resulted in a more pronounced Runge effect, as depicted below.

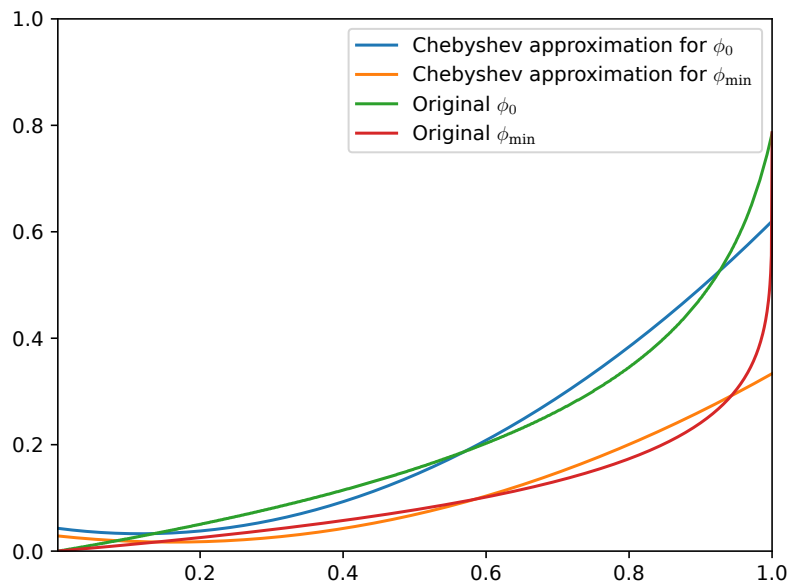


Figure 7: Chebyshev polynomial approximations

5.3 Attempted approximation using exponential function

By far the most accurate fit before the original polynomial approximation, an exponential function of the form $\phi = a \exp(b(\delta - c)) + d$ was used. Significant Runge effect is still evident in the plot below.

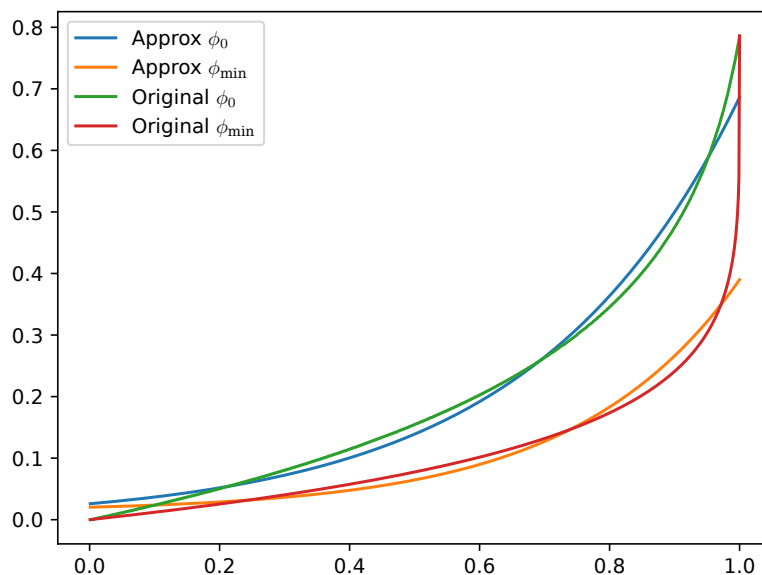


Figure 8: Exponential approximations

5.4 Attempted elliptical integral approximation

Through parameterizing the first-order elliptic integral $K(k; n)$ for some $n \in \mathbb{R}$ and $k \in [0, 1]$ for both translation and dilation control, curve fitting was attempted:

$$K(k; n) = -\frac{\pi}{2n} + \int_0^1 \frac{dx}{n\sqrt{(1-x^2)(1-k^2x^2)}} \quad (14)$$

The optimization routine returned a value of $n = 1$ for both ϕ_0 and ϕ_{\min} , indicating that the first-order approximation is not effective. The plot is shown below.

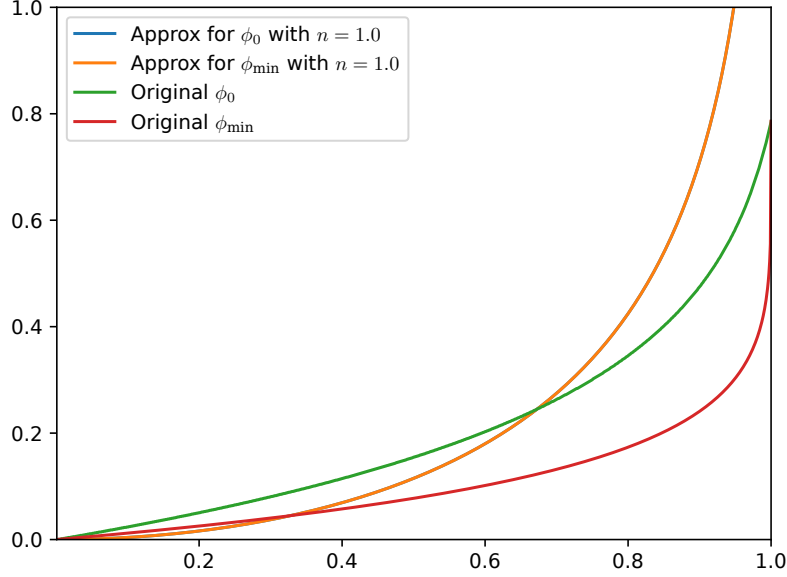


Figure 9: Elliptical approximations

Other possible parameterizations of the first-order functions and possibly of the higher orders will be evaluated, so this is a work-in-progress.

6 Simplifying the expression $1 - w_p(\phi)^2$ from $y_p(\phi)$

Objective: obtain the simplest expression for $1 - w_p(\phi)^2$ in terms of ϕ, β .

The original expression for $1 - w_p(\phi)^2$ is below, with the denominator removed for simplicity.

$$(3\pi + 8)^2 \beta^4 \sin^2(2k)(1 - w_p(\phi)^2) = \underbrace{(3\pi + 8)^2 \beta^4 \sin^2(2k) - [(3\pi + 8)^2 \beta^4 - 8(3\phi + 2)(3\pi + 8)\beta^2 + 16(3\phi + 2)^2]}_{\gamma_1(\phi)} \quad (15)$$

$$- \underbrace{2 \sin(2k)(4 \sin^2(k) + 6)((3\pi + 8)\beta^2 - 4(3\phi + 2))}_{\gamma_2(\phi)} \quad (16)$$

$$- \underbrace{\sin^2(2k)(16 \sin^4(k) + 48 \sin^2(k) + 36)}_{\gamma_3(\phi)} \quad (17)$$

$$= \gamma_1(\phi) - \gamma_2(\phi) - \gamma_3(\phi) \quad (18)$$

Applying Equations 1-3 to the denoted sub-functions $\gamma_1(\phi), \gamma_2(\phi), \gamma_3(\phi)$, we get

$$\gamma_1(\phi) = (3\pi + 8)^2 \beta^4 (\cos^2(2\phi) - 1) + 8(3\phi + 2) ((3\pi + 8)\beta^2 - 2(3\phi + 2)) \quad (19)$$

$$\gamma_2(\phi) = -2(8 \cos(2\phi) + 2 \sin(2\phi) \cos(2\phi)) ((3\pi + 8)\beta^2 - 4(3\phi + 2)) \quad (20)$$

$$= -4 \cos(2\phi) (\sin(2\phi) + 2) ((3\pi + 8)\beta^2 - 4(3\phi + 2)) \quad (21)$$

$$\gamma_3(\phi) = -\cos^2(2\phi) \left(16 \sin\left(\phi + \frac{\pi}{4}\right)^4 + 48 \sin^2\left(\phi + \frac{\pi}{4}\right) + 36 \right) \quad (22)$$

$$= -\cos^2(2\phi) (4 \sin^2(2\phi) + 6)^2 \quad (23)$$

Putting this together, a simplified expression is

$$1 - w_p(\phi)^2 = \frac{\gamma_1}{(3\pi + 8)^2 \beta^4 \cos^2(2\phi)} + \frac{\gamma_2}{(3\pi + 8)^2 \beta^4 \cos^2(2\phi)} + \frac{\gamma_3}{(3\pi + 8)^2 \beta^4 \cos^2(2\phi)} \quad (24)$$

$$= \underbrace{\frac{\cos^2(2\phi) - 1}{\cos^2(2\phi)}}_{\frac{-(1 - \cos^2(2\phi))}{\cos^2(2\phi)} = \frac{-\sin^2(2\phi)}{\cos^2(2\phi)} = -\tan^2(2\phi)} + \frac{8(3\phi + 2)((3\pi + 8)\beta^2 - 2(3\phi + 2))}{(3\pi + 8)^2 \beta^4 \cos^2(2\phi)} \quad (25)$$

$$- \frac{4(\sin(2\phi) + 2)((3\pi + 8)\beta^2 - 4(3\phi + 2))}{(3\pi + 8)^2 \beta^4 \cos(2\phi)} - \frac{(4 \sin^2(2\phi) + 6)^2}{(3\pi + 8)^2 \beta^4} \quad (26)$$

Because $y_p(\phi) = \sqrt{1 - w_p(\phi)^2} \sqrt{\frac{\mu}{r_0}} \cot(\phi + \pi/4)$, by substitution

$$y_p(\phi) = \sqrt{-\tan^2(2\phi) + \frac{8(3\phi + 2)((3\pi + 8)\beta^2 - 2(3\phi + 2))}{(3\pi + 8)^2 \beta^4 \cos^2(2\phi)} - \frac{4(\sin(2\phi) + 2)((3\pi + 8)\beta^2 - 4(3\phi + 2))}{(3\pi + 8)^2 \beta^4 \cos(2\phi)} - \frac{(4 \sin^2(2\phi) + 6)^2}{(3\pi + 8)^2 \beta^4}} \sqrt{\frac{\mu}{r_0}} \cot(\phi + \pi/4) \quad (27)$$