Thrust Normal to Velocity Vector

July 10, 2021

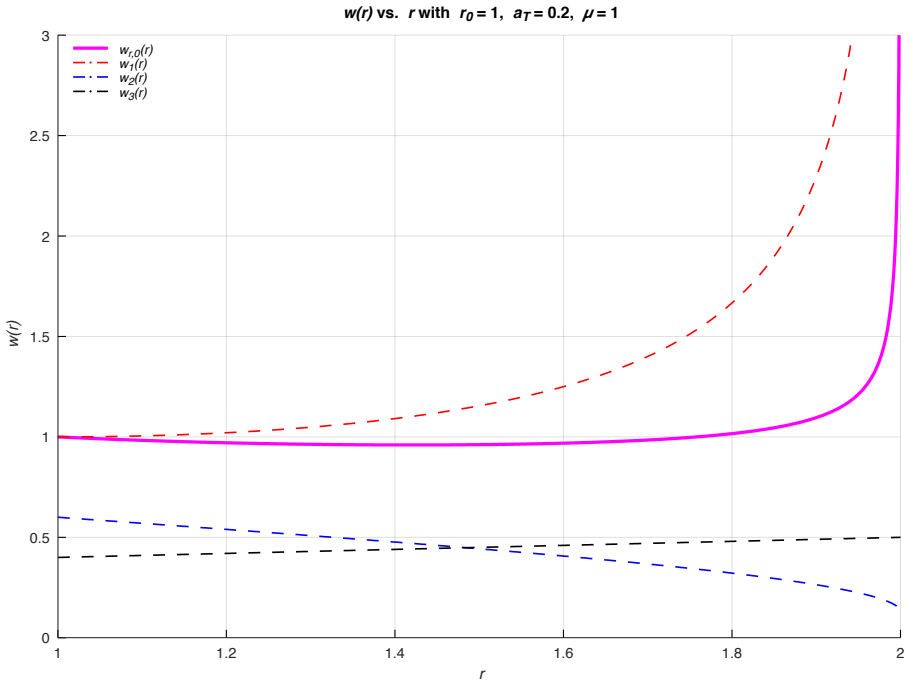
Expressing $w(r)$ in terms of ϕ and β

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 $w_1(r)$

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

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

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

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

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

Reexpressing $w_2(r)$

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

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

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

Reexpressing $w_3(r)$

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

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

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

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

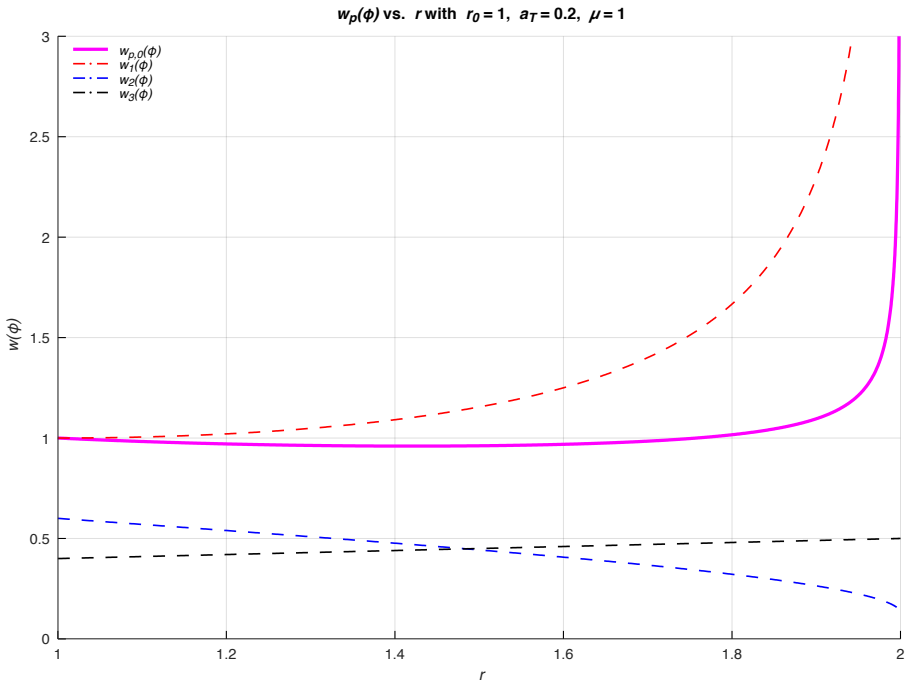
Obtain 11 by $\beta = \sqrt{\frac{4\mu}{(3\pi+8)a_T r_0^2}} \implies \frac{1}{\beta^2} = \frac{(3\pi+8)a_T r_0^2}{4\mu}$

Expression for $w_p(\phi)$

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

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

Checking answer with plots



Expressing $y(r)$ in terms of ϕ and β

Approach similar to $w(r)$:

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

Reexpressing $y_1(r)$

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

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

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

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

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,

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

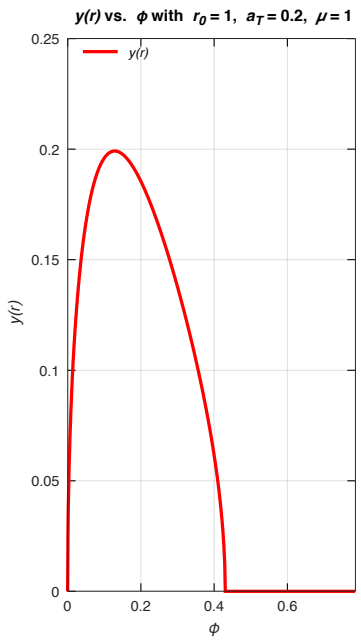
$$- \frac{2\sin(2k)(4\sin^2(k)+6)((3\pi+8)\beta^2 - 4(3\phi+2))}{(3\pi+8)^2\beta^4 \sin^2(2k)} \tag{19}$$

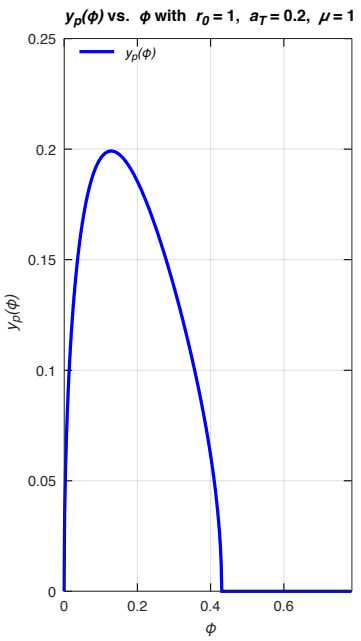
$$- \frac{\sin^2(2k)(16\sin^4(k) + 48\sin^2(k) + 36)}{(3\pi+8)^2\beta^4 \sin^2(2k)} \tag{20}$$

Extracting the denominator,

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

Plots to verify $y_p(\phi)$





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

First derivative test on $w(r)$

$$
\begin{aligned}
{w_1}'(r) &= \frac{d}{dr} r_0 (2r_0 r - r^2)^{-1/2} && (22)\\
&= -\frac{1}{2} r_0 (2r_0 - 2r)(2r_0 r - r^2)^{-3/2} && (23)\\
&= \frac{-r_0(r_0 - r)}{(2r_0 r - r^2)^{3/2}} && (24)\\
{w_2}'(r) &= \frac{-3a_T r_0^2}{\mu} \frac{d}{dr} \arcsin \sqrt{\frac{r}{2r_0}} && (25)\\
&= \frac{-3a_T r_0^2}{\mu} \frac{\frac{1}{4r_0}(\frac{r}{2r_0})^{-\frac{1}{2}}}{\sqrt{1 - \frac{r}{2r_0}}} && (26)\\
&= \frac{-3a_T r_0}{\mu} \frac{1}{4\sqrt{\frac{r}{2r_0}(\frac{2r_0 - r}{2r_0})}} && (27)\\
&= -\frac{3a_T r_0}{2\mu\sqrt{2r_0 r - r^2}} && (28)\\
{w_3}'(r) &= \frac{d}{dr} \frac{a_T r_0}{2\mu}(r + 3r_0) && (29)\\
&= \frac{a_T r_0}{2\mu} && (30)
\end{aligned}
$$

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

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

$$= w_1{}'(r)w_2(r) + w_1(r)w_2{}'(r) + w_3{}'(r) \tag{32}$$

Substituting expressions found prior,

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

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

Reubstituting with $r(\phi)$ and simplifying

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

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

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

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

Then,

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

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

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

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

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 $\phi\in[0,\pi/4)$.

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

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

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

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 $w_p(\phi)$

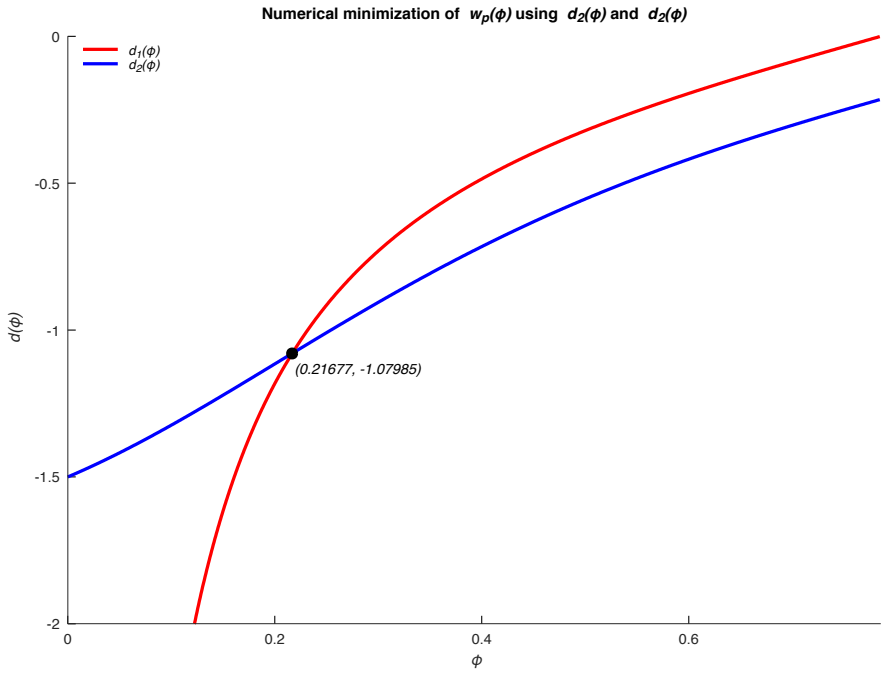
Code:

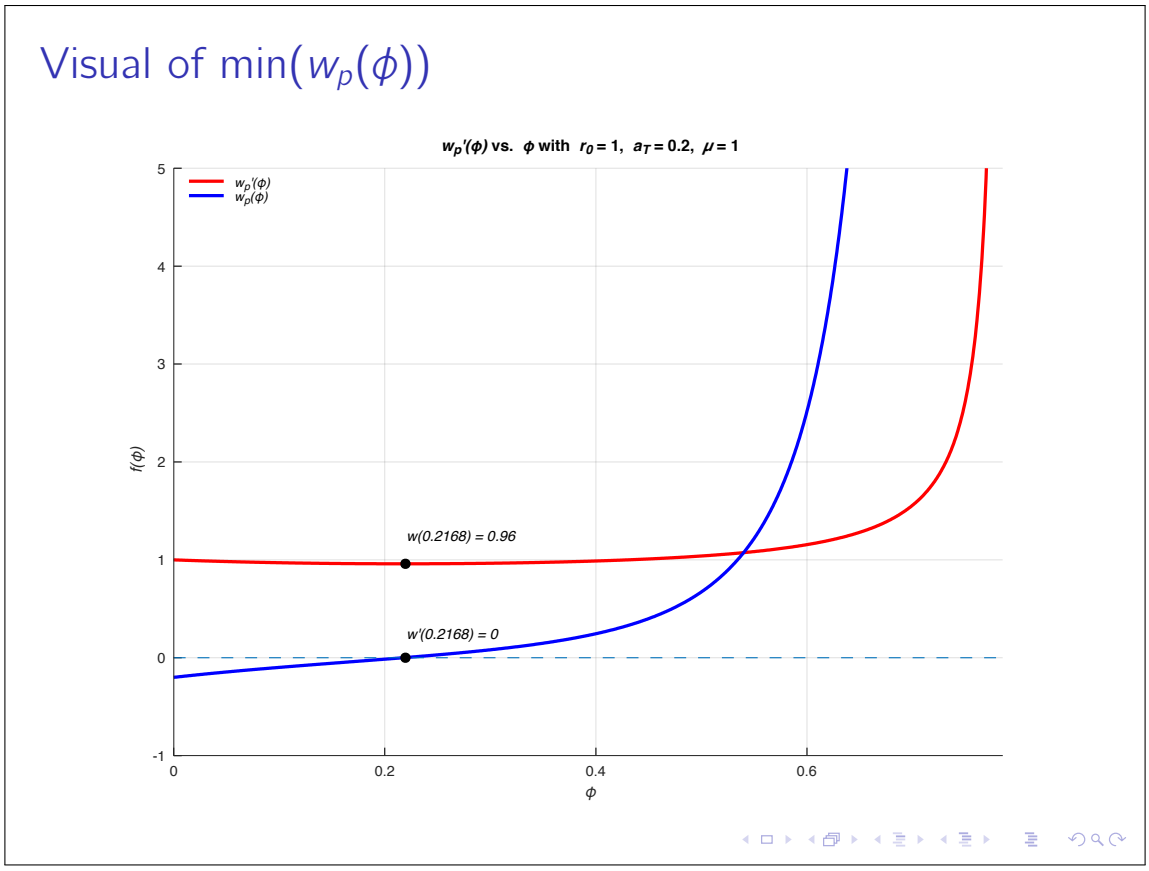
```
d1 = tan(2*k)/(2*mu);
d2 = (1-a_T*r_0^2/mu * (3*phi+2))./(a_T*r_0^2*(sin(2*k).^2-3));

% 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
```

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





1 Simplifying expression for $w_p(\phi)$

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

$$\sin(2k) = \sin(2\phi + \pi/2) = \cos(2\phi). \tag{1}$$

It is also given that

$$2\sin^2(2k) = 1 - \cos(2k). \tag{2}$$

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

$$2\sin^2(2k) = 1 + \sin(2\phi). \tag{3}$$

Finally

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

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

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

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

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

$\varphi(\phi)$ 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_m solves the problem, where we expect `y_minValue = 0` and `phi_m` to be the corresponding value of ϕ:

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

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

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_m changes with respect to it:

```
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 $\vec{\phi_m}$ over $\vec{\delta}$

In Figure 1 above, the vector `y_0` is updated with the value $y_p(\phi_{m,i})$ in each iteration for each value

of β, and is expected to be 0. The vector `w_1` is updated with $w_p(\phi_{m,i})$, which is expected to be unity. As was done with `y_p(Beta)`, we have defined a vector `w_p(Beta)` to represent $w_p(\phi)$ where `Beta` is a constant that can be varied. Thus, we should expect $w_p(\vec{\phi}_m) = \vec{1}$, $y_p(\vec{\phi}_m) = \vec{0} \in \mathbb{R}^n$, 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 `y_p(Beta_vec(i)` is accessed to find $\phi_{m,i}$ 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. `length(y_p(Beta_vec(i))` is 1), so we are only able to find $\phi_{m,i} = 0 \implies y(\phi_m) = 0$.

The results of ϕ_m vs. δ are expressed in Figure 2 below. $y_p(\phi_m) \approx 0$ and $w_p(\phi_m) \approx 1$ within numerical error, which verifies the validity of these results.

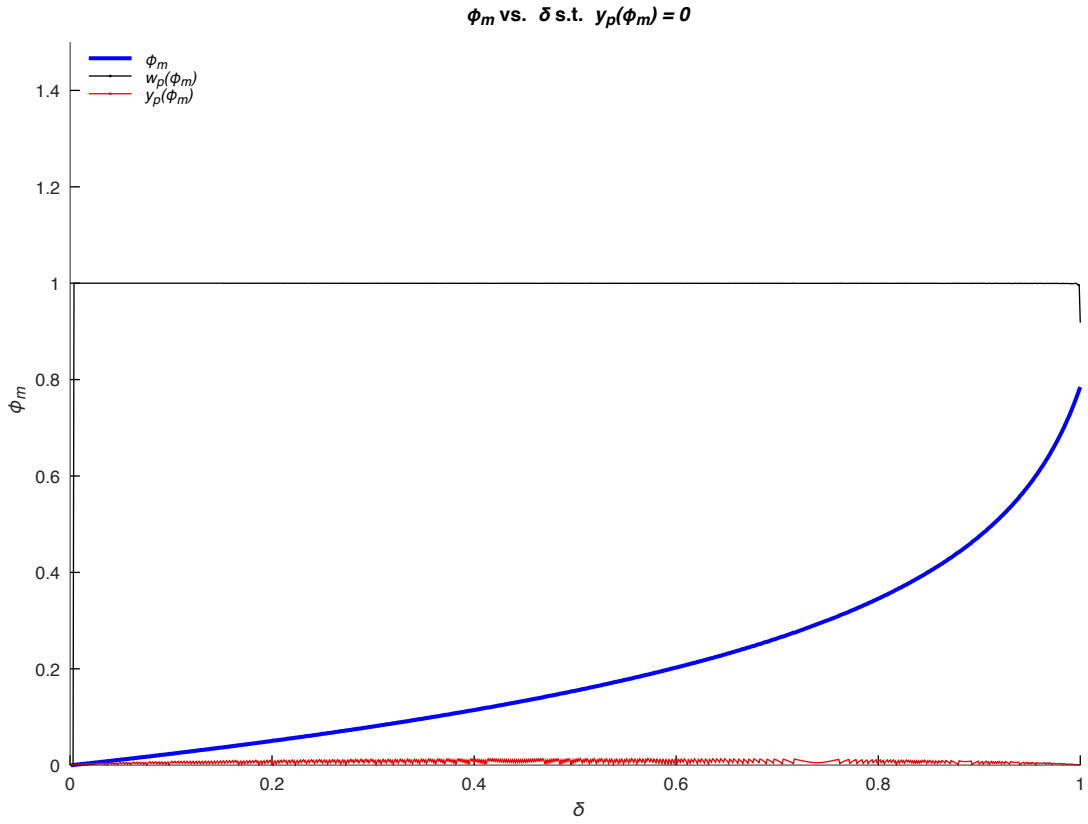


Figure 2: A plot of these results

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 $w_p(\phi_{m,i})$ where $\phi = \phi_{m,i}$ 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 $\phi_{m,i}$ 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 $\vec{\phi}_m$ for $\min(w_p(\phi))$ for various β

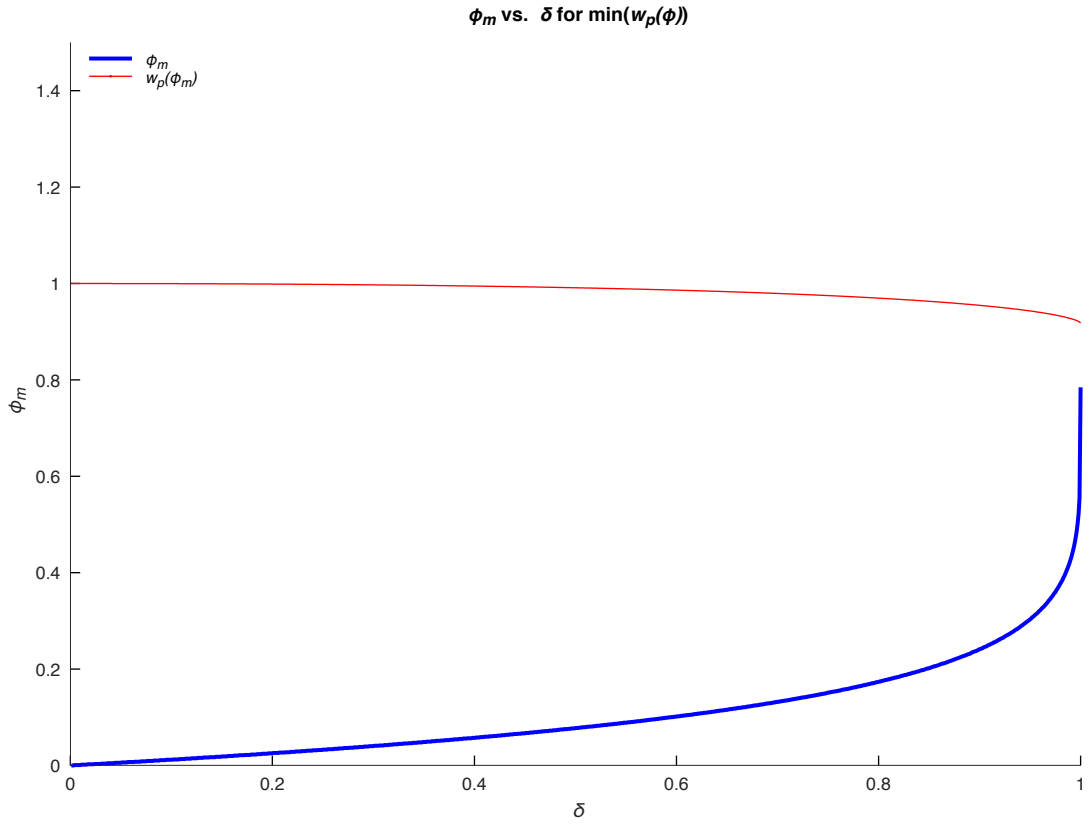


Figure 4: Values of ϕ_m that minimize $w_p(\phi)$ for various β

Figure 4, showing ϕ_m (representing `phi_m2`) vs. δ, summarizes these results.

4 Bisection and Newton-Raphson method for finding roots

4.1 Bisection method

Let $f(x) \in \mathbb{R}$ be continuous and $x, c \in [a, b]$. Also define $\epsilon_0 = |a - b|$ as the initial error. The approach to the bisection method involves using a finite number of iterations where the original

domain interval $[a, b]$ (and likewise the error) is continuously halved until the desired tolerance ϵ and thusly the root of the function in question is found. From, this we can conclude that the required precision/tolerance is $\epsilon = |c_n - c|$ where $f(c) = 0$ and c_n is the upper or lower bound of the interval at the nth iteration. Finding an upper bound for the value of n eliminates the need for an infinite loop and prevents divergence of the bisection algorithm. The upper bound for n is expressed as [1]

$$|c_n - c| \leq \frac{|b-a|}{2^n} \implies n \leq \log(\frac{\epsilon_0}{\epsilon}). \tag{8}$$

We will use the bisection method to find the roots for $w_p{}'(\phi)$ in order to minimize $w_p(\phi)$ and find when $y_p(\phi) = 0$ for a given tolerance ϵ.

4.1.1 Minimizing $w_p(\phi)$

4.1.2 Roots for $y_p(\phi)$

4.2 Newton-Raphson method

The implementation of the Newton-Raphson algorithm used here was derived from the pseudocode on the corresponding Wikipedia article[2].

The code used to implement both algorithms is located here.

5 Curve fitting

[1] Wikipedia - Bisection method
[2] Wikipedia - Newton's method