
UMERC+METS
2024 Conference

*7-9 August | Duluth, MN, USA*

# Open-source toolbox for semi-analytical hydrodynamic coefficients via the matched eigenfunction expansion method

Rebecca McCabe[a*], Kapil Khanal[b], Maha Haji[ab]

[a]*Sibley School of Mechanical Engineering, Cornell University, 124 Hoy Rd, Ithaca NY 14853, USA*
[b]*Department of Systems Engineering, Cornell University, 602 Rhodes Hall, Ithaca NY 14853, USA*

**Abstract**

Floating body hydrodynamics are typically solved numerically using the boundary element method. The associated code is computationally costly, scaling with the number of mesh panels, and can have accuracy issues at specific frequencies and for thin bodies. In this work, we instead implement a previously-developed matched eigenfunction expansion method to semi-analytically solve the linear potential flow radiation problem for axisymmetric bodies. This method first establishes distinct fluid regions based on the body geometry and expresses the velocity potential as a function of vertical and radial basis functions (eigenfunctions) with unknown coefficients. Eigenfunctions are chosen to automatically enforce several boundary conditions of the problem. The coefficients are found by truncating and solving an infinite linear system representing the matching of potential and radial velocity across fluid region boundaries. This yields a solution for the 3D potential and the hydrodynamic coefficients. We compare the results and computational complexity of the matched eigenfunction expansion method with that of the standard boundary element method. Benefits of the former include 10x faster solve time and lack of meshing, which are particularly appealing in optimization workflows. Our framework will be released as an open-source python package to enable future integration with design tools, implementation of gradients, and democratization of this efficient method. This is a meaningful contribution because prior relevant implementations of the matched eigenfunction expansion method are, to the authors' knowledge, private and not available open-source or even commercially. Future work will extend this formulation to different kinds of bodies and arrays.

*Keywords:* semi-analytical hydrodynamics; matched eigenfunction expansion method; numerical methods; open-source software

## 1. Introduction

Semi-analytical solutions to hydrodynamic boundary value problems can outperform numerical solutions in both accuracy and computationally cost, but implementation complexity and a lack of available solvers limits their practical use. This work replicates an existing semi-analytical model and releases the code open source in MATLAB and Python. We discuss implementation details which the original publications omit, propose a roadmap for extensions, and provide an implementation to enable future benchmarking and application of the method.

* Corresponding author

## 2. Methods

### *2.1. Linear Hydrodynamics and Eigenfunctions*

The dynamics of a floating body in water waves are well-described by linear potential flow theory, a simplification of the Navier-Stokes equation. This theory states that the fluid velocity field $\vec{v}$ is the gradient of some complex potential $\phi, \vec{v} = \nabla\phi$, and $\phi$ satisfies the Laplace equation, $\nabla^2\phi = 0$. Adding the free surface condition, far-field or incident waves, and body surface and sea-bed conditions detailed in [1] yields a boundary value problem. When boundary conditions correspond to the heave radiation problem (body moving vertically, no incident waves), solving for the potential $\phi(r, \theta, z)$ determines the heave added mass and damping $\mu_{33}$ and $\lambda_{33}$, hereafter "hydro coefficients":

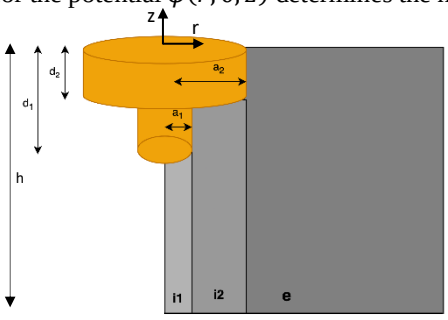


*Figure 1: fluid domain and body dimensions*

$$\mu_{33} + \frac{i\lambda_{33}}{\omega} = \rho h^3 \iint_\Gamma \phi \frac{\partial \phi}{\partial z} dA \tag{1}$$

For appropriate geometries, the PDE is separable and $\phi$ can be expressed as the product of radial, vertical, and circumferential basis functions called eigenfunctions. In cylindrically symmetric problems, the radial eigenfunctions are a family of transcendental functions called Bessel functions. The fluid is then divided into cylindrical regions. Arbitrarily many fluid regions can exist, so the method applies to any axisymmetric geometry, including multiple concentric bodies that oscillate independently. Here, two concentric cylinders and thus three fluid regions are demonstrated. Extension to many regions is discussed in section 3.4. Figure 1 illustrates the regions and dimensions: two internal regions *i1* and *i2*, and an external region *e* extending to infinity.

The potential in each region is split into a homogeneous part for the unforced solution and a particular part due to body motion: $\phi = \phi_h + \phi_p$. Boundary conditions dictate $\phi_p$ and the eigenfunctions for $\phi_h$ in each region, which the textbook [1] describes in detail. Table 1 shows the equations originally presented in [2], [3] for the potential and eigenfunctions in each region, which include infinitely many unknown eigencoefficients $C_{1n}^{i1}, C_{1m}^{i2}, C_{2m}^{i2}$, and $B_k^e$. By construction, this potential obeys all boundary conditions except for zero radial velocity on radial body surfaces. The unknown coefficients must be computed to enforce this final condition as well as continuity across regions, which will be the subject of section 2.2.

Table 1. Equations for potential (homogeneous and particular) and eigenfunctions (radial and vertical) for each region.

| Region | i1 | i2 | e |
|---|---|---|---|
| Homog. potential $\phi_h(r,z)$ | $\sum_{n=0}^{\infty} C_{1n}^{i1} R_{1n}^{i1}(r) Z_n^{i1}(z)$ | $\sum_{m=0}^{\infty} \left(C_{1m}^{i2} R_{1m}^{i2}(r) + C_{2m}^{i2} R_{2m}^{i2}(r)\right) Z_m^{i2}(z)$ | $\sum_{k=0}^{\infty} B_k^e \Lambda_k(r) Z_k^e(z)$ |
| Partic. potential $\phi_p(r,z)$ | $\begin{cases} \frac{1}{2(h-d_1)}\left[(z+h)^2 - \frac{r^2}{2}\right], & 1M \\ 0, & 1S \end{cases}$ | $\begin{cases} \frac{1}{2(h-d_2)}\left[(z+h)^2 - \frac{r^2}{2}\right], & 2M \\ 0, & 2S \end{cases}$ | 0 |
| Radial eigen-functions | $R_{1n}^{i1}(r) = \begin{cases} \frac{1}{2}, & n = 0 \\ \frac{I_0(\lambda_n^{i1} r)}{I_0(\lambda_n^{i1} a_2)}, & n \geq 1 \end{cases}$ | $R_{1m}^{i2}(r) = \begin{cases} \frac{1}{2}, & m = 0 \\ \frac{I_0(\lambda_m^{i2} r)}{I_0(\lambda_m^{i2} a_2)}, & m \geq 1 \end{cases}$<br>$R_{2m}^{i2}(r) = \begin{cases} \frac{1}{2}\ln\left(\frac{r}{a_2}\right), & m = 0 \\ \frac{K_0(\lambda_m^{i2} r)}{K_0(\lambda_m^{i2} a_2)}, & m \geq 1 \end{cases}$ | $\Lambda_k(r) = \begin{cases} \frac{H_0^1(m_0 r)}{H_0^1(m_0 a_2)}, & k = 0 \\ \frac{K_0^1(m_k r)}{K_0^1(m_k a_2)}, & k \geq 1 \end{cases}$ |
| Vertical eigen-function | $Z_n^{i1}(z) = \begin{cases} 1, & n = 0 \\ \sqrt{2}\cos(\lambda_n^{i1}(z+h)), & n \geq 1 \end{cases}$ | $Z_m^{i2}(z) = \begin{cases} 1, & m = 0 \\ \sqrt{2}\cos(\lambda_m^{i2}(z+h)), & m \geq 1 \end{cases}$ | $Z_k^e(z) = \begin{cases} N_0^{1/2}\cosh(m_0(z+h)), & k = 0 \\ N_k^{1/2}\cos(m_k(z+h)), & k \geq 1 \end{cases}$ |

Variables in Table 1 are: $\lambda_j^{iq} = \frac{j\pi}{h-d_q}$ for j={n,m}, q={1,2}; $m_0 \tanh(m_0 h) = \omega^2/g$; $m_k \tan(m_k h) = -\omega^2/g, k \geq 1$; $N_0 = \frac{1}{2}\left(1 + \frac{\sinh(2m_0 h)}{2m_0 h}\right)$; $N_k = \frac{1}{2}\left(1 + \frac{\sin(2m_k h)}{2m_k h}\right), k \geq 1$. $I_0$, $K_0$, and $H_0^1$ are different Bessel functions of order zero; and 1M and 2M mean body 1 and 2 are moving respectively, while 1S and 2S mean each is stationary.

### *2.2. Matching Across Fluid Boundaries*

The eigencoefficients must be selected to enforce the radial velocity body boundary condition and the matching of the potentials and radial velocities at the edges of each region, earning this technique the name Matched Eigenfunction Expansion Method (MEEM). The radiation problem was first solved this way for a floating cylinder in 1980 [4].

First, the infinite sums in $\phi_h$ must be truncated. Assuming truncation to $N$ terms in *i1,* $M$ terms in *i2*, and $K$ terms in *e*, the total number of eigencoefficients to solve for is $N+2M+K$. For a 3-region problem, there are 2 boundaries. Thus there are four matching equations: (1) potential at $a_1$, (2) potential at $a_2$, (3) velocity at $a_1$, and (4) velocity at $a_2$. As-is, this is not enough equations ($4 < N+2M+K$). We must leverage eigenfunction orthogonality to get enough equations. The first equation will turn into N equations; the second and third each give M; the fourth K. The transformation uses the following property of orthogonality. Consider a generic function Y(x) expressed as a series with coefficients $\alpha$ and basis functions $e(x)$: $Y(x) = \sum_i \alpha_i e_i(x)$. If $e_j(x)$ is orthogonal to $e_i(x)$ from $x = a$ to $b$, then:

$$\int_a^b Y(x)e_j(x)dx = (b-a) < Y, e_j >= (b-a) < \sum_i \alpha_i e_i, e_j >= (b-a)\sum_i \alpha_i < e_i, e_j >= (b-a)\sum_i \alpha_i \delta_{ij} = (b-a)\alpha_j$$

where $<\cdot,\cdot>$ is inner product and $\delta_{ij}$ is Kronecker's delta. In the current hydrodynamics problem, the basis functions are the vertical eigenfunctions $Z_n^{i1}, Z_m^{i2}$, and $Z_k^e$. Orthogonality of each eigenfunction can be verified with the inner product. In the first region, for example, $< Z_{n_1}^{i1}, Z_{n_2}^{i1} >= \delta_{n_1 n_2}$. Note that eigenfunctions of different domains are not orthogonal, and their inner products will be expressed as coupling integrals in section 2.3.

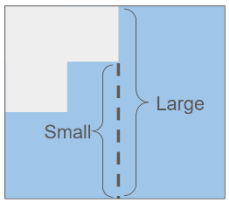

*Figure 2: small and large fluid regions*

For each of the four matching equations, the property of orthogonality applies only after multiplying by the appropriate eigenfunction and integrating over appropriate bounds. For the potential matching equations, multiply both sides by the eigenfunction of the region with smaller fluid height (so $Z_n^{i1}$ at $a_1$ and $Z_m^{i2}$ at $a_2$). Then integrate over that fluid height (z=-h to $-d_1$ at $a_1$, and -h to $-d_2$ at $a_2$). For velocity matching, multiply instead by the eigenfunction corresponding to the larger region, while still integrating over the smaller region. In velocity matching, an extra step is required to incorporate the boundary condition of zero radial velocity along the radial surface of the body. Since it is zero-valued, the integral of this velocity may be added to one side of the equation (the one corresponding to the velocity of the larger region) to change the integration bounds only on that side. This manifests in the bounds of the coupling integrals to be presented in section 2.3. Other combinations of eigenfunction multiplication or integration besides those described above are not useful since they result in integrating a quantity on a region where it is undefined, or a form unsuitable for the application of the orthogonality property.

### *2.3. Block Matrix Structure*

Once orthogonality is applied, the matching equations create a linear system $A\vec{x} = \vec{b}$ where $A$ is a complex sparse $(N+2M+K)$ x$(N+2M+K)$ square matrix corresponding to the homogeneous case, $\vec{x} = \left[\overrightarrow{C_{1n}^{i1}}, \overrightarrow{C_{1m}^{i2}}, \overrightarrow{C_{2m}^{i2}}, \overrightarrow{B_k^e}\right]$ is the complex eigencoefficient vector, and $\vec{b}$ is the real boundary condition vector corresponding to the particular case. We elaborate on the block structure of the A-matrix and b-vector, an implementation detail that prior discussion of MEEM overlooks. The A-matrix and b-vector block structures are shown in Table 2 and 3 respectively. They are written in compact notation using row vectors of basis functions, so $\overrightarrow{R_{1n}^{i1}} = \left[R_{10}^{i1}, R_{11}^{i1}, \ldots, R_{1(N-1)}^{i1}\right]$ and so on. Each basis function is evaluated at the radius described to the left of its row in the table. $0_{ij}$ and $1_{ij}$ are the $i$ x$j$ matrices of zeros and ones respectively; diag(·) constructs a diagonal matrix from a vector; and $\odot$ is the Hadamard (element-wise) product. Of the sixteen blocks that make up the matrix, six are diagonal, four are zero, and six are dense, resulting in the sparsity pattern shown in Figure 3. The dense blocks contain coupling integrals $I$ of the vertical eigenfunctions:

$$I_{nm} = I_{mn}^T = \int_{-h}^{-d_1} \overrightarrow{Z_n^{i1}}^T \overrightarrow{Z_m^{i2}} dz, \qquad I_{mk} = I_{km}^T = \int_{-h}^{-d_2} \overrightarrow{Z_m^{i2}}^T \overrightarrow{Z_k^e} dz \tag{2}$$

These coupling integrals are definite integrals of cos and cosh, and can be solved analytically with symbolic math. Once the linear system is solved for $\vec{x}$, obtain potential from Table 1, then plug into (1) to get the hydro coefficients.

Table 2. A-matrix

| | | | $\overrightarrow{C_{1n}^{i1}}$ | $\overrightarrow{C_{1m}^{i2}}$ | $\overrightarrow{C_{2m}^{i2}}$ | $\overrightarrow{B_k^e}$ |
|---|---|---|---|---|---|---|
| Equation | r | size | N | M | M | K |
| $\phi^{i1} = \phi^{i2}$ | $a_1$ | N | $(h-d_1)\,\mathrm{diag}\left(\overrightarrow{R_{1n}^{i1}}\right)$ | $-I_{nm} \odot 1_{N1}\overrightarrow{R_{1m}^{i2}}$ | $-I_{nm} \odot 1_{N1}\overrightarrow{R_{2m}^{i2}}$ | $0_{NK}$ |
| $\phi^{i2} = \phi^{e}$ | $a_2$ | M | $0_{MN}$ | $(h-d_2)\,\mathrm{diag}\left(\overrightarrow{R_{1m}^{i2}}\right)$ | $(h-d_2)\,\mathrm{diag}\left(\overrightarrow{R_{2m}^{i2}}\right)$ | $-I_{mk} \odot 1_{M1}\overrightarrow{\Lambda_k}$ |
| $\frac{\partial}{\partial r}\phi^{i1} = \frac{\partial}{\partial r}\phi^{i2}$ | $a_1$ | M | $-I_{mn} \odot 1_{M1}\frac{\partial}{\partial r}\overrightarrow{R_{1n}^{i1}}$ | $(h-d_2)\,\mathrm{diag}\left(\frac{\partial}{\partial r}\overrightarrow{R_{1m}^{i2}}\right)$ | $(h-d_2)\,\mathrm{diag}\left(\frac{\partial}{\partial r}\overrightarrow{R_{2m}^{i2}}\right)$ | $0_{MK}$ |
| $\frac{\partial}{\partial r}\phi^{i2} = \frac{\partial}{\partial r}\phi^{e}$ | $a_2$ | K | $0_{KN}$ | $-I_{km} \odot 1_{K1}\frac{\partial}{\partial r}\overrightarrow{R_{1m}^{i2}}$ | $-I_{km} \odot 1_{K1}\frac{\partial}{\partial r}\overrightarrow{R_{2m}^{i2}}$ | $h\,\mathrm{diag}\left(\frac{\partial}{\partial r}\overrightarrow{\Lambda_k}\right)$ |

Table 3. b-vector

| | |
|---|---|
| N | $\int_{-h}^{-d_1}(\phi_p^{i2} - \phi_p^{i1})\overrightarrow{Z_n^{i1\,T}}\,dz$ |
| M | $-\int_{-h}^{-d_2}\phi_p^{i2}\overrightarrow{Z_m^{i2T}}\,dz$ |
| M | $\int_{-h}^{-d_1}\frac{\partial}{\partial r}\phi_p^{i1}\overrightarrow{Z_m^{i2T}}dz - \int_{-h}^{-d_2}\frac{\partial}{\partial r}\phi_p^{i2}\overrightarrow{Z_m^{i2T}}dz$ |
| K | $\int_{-h}^{-d_2}\frac{\partial}{\partial r}\phi_p^{i2}\overrightarrow{Z_k^{eT}}\,dz$ |

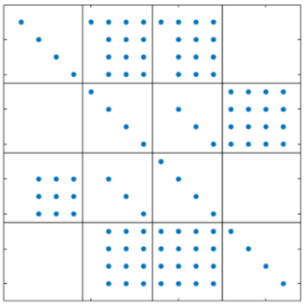

*Figure 3: A-matrix sparsity pattern, shown for N=M=K=4.*

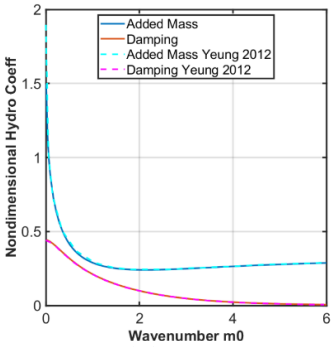


*Figure 4: validation against [3] MEEM solution in shallow water*

## 3. Results

### 3.1. Validation

Hydro coefficient results are validated by comparing to a benchmark shallow-water concentric-cylinder MEEM solution in [3]. Excellent agreement is observed, shown in Figure 4. Authors of [3] also experimentally validate their results in [5]. Then, results are compared to WAMIT results for the RM3 point absorber in deep water. RM3 is not a perfect concentric cylinder, since the float has a conical bottom and the spar has a damping plate. The N=M=K=11 case is shown in Figure 5. Reasonable agreement is observed, with the greatest error at low frequencies.

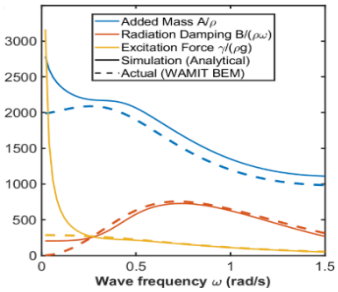


*Figure 5: validation against WAMIT for RM3 geometry in deep water*

### 3.2. Convergence and Other Numerical Notes

As $N, M, K \to \infty$, matching quality improves, and hydro coefficients converge toward their true values. Previous MEEM papers use N=M=K=50 to obtain 4-digit matching accuracy without elaborating on convergence properties [3]. We observe that potential matching converges faster than velocity matching. Figure 6 shows the matching behavior for N=M=K=11, where potential matches well but velocity still has noticeable mismatch. Hydro coefficient convergence depends on the geometry: the benchmark shallow-water geometry of [3] converges to within 0.25% with only N=M=K=4, but RM3 requires N=M=K>10, shown in figure 7. There, damping converges well at low frequencies but requires more harmonics at higher frequencies, while added mass has similar convergence across frequencies. At high N or M, the A-matrix is ill-conditioned since modified Bessel functions of the first kind $I_0$ grow

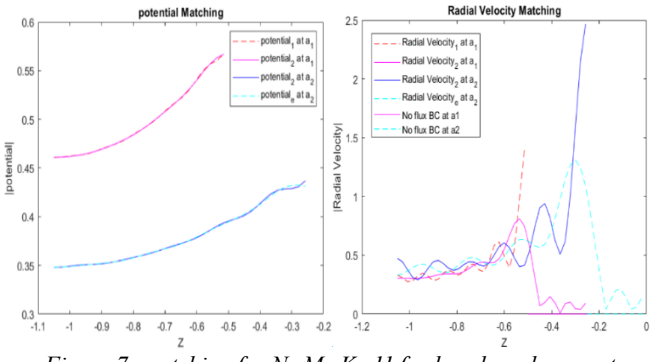


*Figure 7: matching for N=M=K=11 for benchmark geometry*

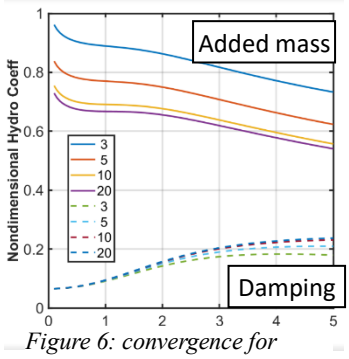


*Figure 6: convergence for N=M=K={3,5,10,20} for RM3*

exponentially. Condition numbers of >1e14 were observed. Scaling the highest rows of the A-matrix and b-vector reduced the condition number by an order of magnitude and minimally affected eigencoefficients, so the result was deemed tentatively acceptable with further investigation warranted. A final numerical subtlety worth discussing is finite precision effects in calculating $m_k$. Bounds of $180^o \cdot \left[k - \frac{1}{2}, k\right]$ are placed on $m_k h$ in a root-finding algorithm to ensure the kth root is identified, where degrees are used instead of radians so asymptotes occur at rational values.

### *3.3. Runtime and Computational Cost Scaling*

The runtime of the MEEM method is the time required to find the eigencoefficients, then obtain the hydrodynamic coefficients from eigencoefficients. Generating the A-matrix and b-vector for the eigencoefficients involves running a nonlinear root-finding algorithm K times to generate the $m_k$ inputs, then evaluating Bessel functions 2N+8M+2K times. The cost of evaluating coupling integrals (2) is negligible since they are trigonometric. Linear solves scale almost cubically with matrix size, so this step scales with $(N+2M+K)^3$. Obtaining hydrodynamic coefficients via integral (1) results in new coupling integrals over the radial eigenfunctions. These coupling integrals can be simplified symbolically and require 6M Bessel evaluations to find hydro coefficients for the outer cylinder, and 2N for the inner cylinder [3]. The current implementation instead calculates the radial coupling integrals numerically, which adds overhead, so future work should use analytical solutions. For N=M=K=10, the simulation averages 31 ms on a Windows 10 laptop with a 2.5 GHz Intel i9 processor. Figure 8 shows the time breakdown, split remarkably evenly between computing eigencoefficients and hydro coefficients. For both, the dominant cost is Bessel evaluation, so future code optimization should focus on speeding up Bessel evaluations, such as with lookup tables. [3] proposes using the sparsity pattern to reduce matrix size from N+2M+K to 2M, but this seems low impact since the linear solve only takes 5% of compute time. On the other hand, matrix size in a boundary element method solver is much larger (meshes may have 1000s of panels) and the linear solve drives computation cost. On the same machine, Capytaine boundary element method for the same geometry takes an average of 323 ms for a 710 panel mesh (1% convergence). Thus, MEEM achieves a 10x time reduction over Capytaine.

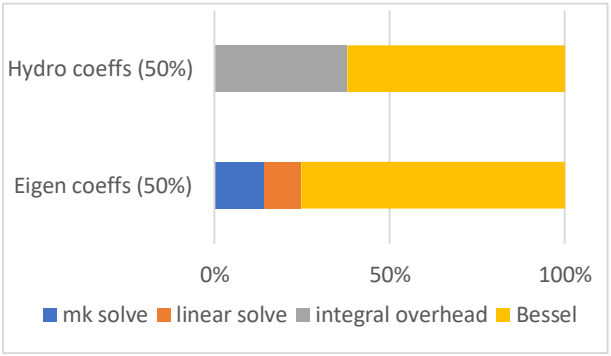


*Figure 8: bar graph showing computational time breakdown*

### *3.4. Discussion and future work*

The method has been implemented in both MATLAB (available with MIT license at https://github.com/symbiotic-engineering/MDOcean/blob/main/mdocean/simulation/modules/MEEM/run_MEEM.m) and Python (to be released open-source soon). The MATLAB version uses symbolic algebra to automatically generate the A-matrix and b-vector from matching equations, then substitutes geometric values and solves numerically. Symbolics were chosen over a purely numeric approach to reduce formulation mistakes, expedite future extension to other geometries, and potentially yield analytic gradients. The latter was unsuccessful because on a machine with 64GB RAM, the symbolic engine runs out of memory attempting to differentiate the linear system for any reasonable values of N, M, and K. With the structure of the A-matrix verified in MATLAB, the Python version implements this structure purely numerically to reduce computation time. Based on the open-source nature and growing popularity of Python for hydrodynamics, the Python version is intended as primary, and will be released as a package to facilitate integration with other modules such as WecOptTool. Results presented here were obtained with the MATLAB version.

Future work is required to obtain gradients, for example via adjoints or algorithmic differentiation. While the method is fast enough that a finite difference approach may suffice, the oscillatory nature of Bessel functions can make finite difference inaccurate. Additionally, planned future work involves extending the package to other modes of motion, as in [4], and other geometries. For example, the method can be extended to axisymmetric bodies with arbitrary piecewise-constant profiles, as in [6]. This would enable rapid shape optimization and control co-design.

**Acknowledgements**

We thank Prof. R. W. Yeung and Seung-Yoon Han for discussions on the theory and computation of this method.

**References**

[1] I. K. Chatjigeorgiou, *Analytical Methods in Marine Hydrodynamics*. Cambridge: Cambridge University Press, 2018. doi: 10.1017/9781316838983.

[2] F. P. Chau and R. W. Yeung, "Inertia and Damping of Heaving Compound Cylinders," presented at the 25th International Workshop on Water Waves and Floating Bodies, Harbin, China, Jan. 2010. Accessed: Sep. 27, 2023. [Online]. Available: https://www.academia.edu/73219479/Inertia_and_Damping_of_Heaving_Compound_Cylinders_Fun

[3] F. P. Chau and R. W. Yeung, "Inertia, Damping, and Wave Excitation of Heaving Coaxial Cylinders," presented at the ASME 2012 31st International Conference on Ocean, Offshore and Arctic Engineering, American Society of Mechanical Engineers Digital Collection, Aug. 2013, pp. 803–813. doi: 10.1115/OMAE2012-83987.

[4] R. W. Yeung, "Added mass and damping of a vertical cylinder in finite-depth waters," *Appl. Ocean Res.*, vol. 3, no. 3, pp. 119–133, Jul. 1981, doi: 10.1016/0141-1187(81)90101-2.

[5] D. Son, V. Belissen, and R. W. Yeung, "Performance validation and optimization of a dual coaxial-cylinder ocean-wave energy extractor," *Renew. Energy*, vol. 92, pp. 192–201, Jul. 2016, doi: 10.1016/j.renene.2016.01.032.

[6] K. Kokkinowrachos, S. Mavrakos, and S. Asorakos, "Behaviour of vertical bodies of revolution in waves," *Ocean Eng.*, vol. 13, no. 6, pp. 505–538, Jan. 1986, doi: 10.1016/0029-8018(86)90037-5.