The Journal of Open Source Software

# PathFinder: A Matlab/Octave package for oscillatory integration

**Andrew Gibbs** [1]

**1** University College London, United Kingdom

**DOI:** 10.xxxxxx/draft

**Software**

- Review
- Repository
- Archive

**Editor:** Open Journals

**Reviewers:**

- @openjournals

**Submitted:** 01 January 1970
**Published:** unpublished





## Summary

Oscillatory integrals arise in models of a wide range of physical applications, from acoustics to quantum mechanics. `PathFinder` is a Matlab/Octave package for efficient evaluation of oscillatory integrals of the form

$$I = \int_a^b f(z)\exp(\mathrm{i}\omega g(z))\,\mathrm{d}z, \tag{1}$$

where the endpoints $a$ and $b$ can be complex-valued, even infinite; $\omega > 0$ determines the angular frequency, $f(z)$ is a non-oscillatory entire function and $g(z)$ is a polynomial phase function. The syntax is simple:

```
I = PathFinder(a, b, f, gCoeffs, omega, N);
```

Here, `f` is a function handle representing $f(z)$, `gCoeffs` is a vector of coefficients of $g(z)$, `omega` is the frequency parameter $\omega$ and `N` is a parameter that controls the degree of approximation.

`PathFinder` is the first black-box software that can evaluate (1) accurately, robustly and efficiently for any $\omega > 0$. It will be useful across many scientific disciplines, for problems that were previously too computationally expensive or too mathematically challenging to solve.

## Statement of need

Based on the method of Numerical Steepest Descent (Huybrechs & Vandewalle, 2006), `PathFinder` is an implementation of the algorithm described in Gibbs et al. (2024), where an earlier version of the code was used to produce numerical experiments. Since these experiments, much of the code has been rewritten in C, interfacing with Matlab/Octave via MEX (Matlab executable) functions. These are easily compiled using a single script.

### Ease of use

Standard quadrature rules (midpoint rule, Gauss quadrature, etc) are easy to use, and many open-source implementations are available. However, when applied to (1), such methods become prohibitively inefficient for large $\omega$.

On the other hand, several methods exist for the efficient evaluation of oscillatory (large $\omega$) integrals such as (1); a thorough review is given in Deaño et al. (2018). However, applying these methods often requires an expert understanding of the process and a detailed analysis of the integral, making such methods inaccessible to non-mathematicians. Even with the necessary mathematical understanding, models may require hundreds or thousands of oscillatory integrals to be evaluated, making detailed analysis of each integral highly challenging or impossible.

Despite being based on complex mathematics, `PathFinder` can be easily used by non-mathematicians. The user must simply understand the definitions of the components of (1).

### Use in academic research

In many physical models, interesting physical phenomena occur in the presence of *coalescing saddle points* (see e.g. Gibbs et al. (2024) for a definition). Examples include chemical reactions, rainbows, twinkling starlight, ultrasound pulses, and focusing of sunlight by rippling water (*NIST Digital Library of Mathematical Functions*, 2023, sec. 36.14).

Coalescing saddle points can cause steepest descent methods to break down, even in simple cases where $g(z)$ is a cubic polynomial (Huybrechs et al., 2019). By design, `PathFinder` is robust for any number of coalescing saddle points. This is demonstrated in Figures 1 and 2, where `PathFinder` has been used to model well-known optics problems with coalescing saddle points. Here, each point $(x_1, x_2)$ requires a separate evaluation of (1) and thus a separate call to `PathFinder`.

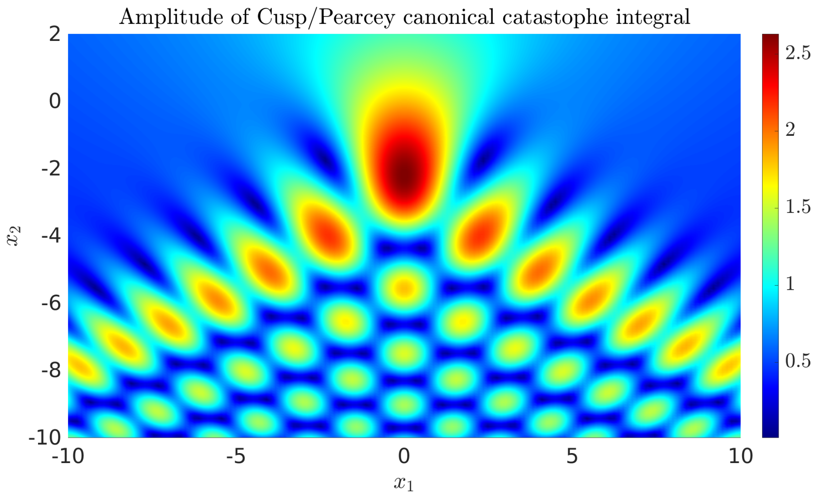


**Figure 1:** PathFinder approximation to Pearcey/Cusp Catastrophe integrals (Pearcey, 1946), which contain coalescing saddle points.

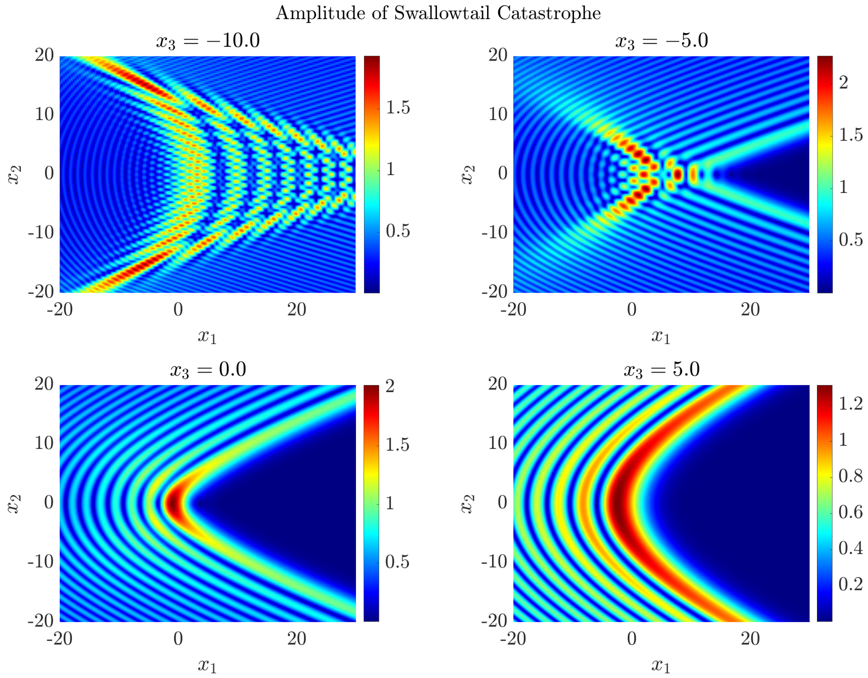


**Figure 2:** PathFinder approximation to Swallowtail Catastrophe integrals (Arnol'd, 1981), which contain many coalescing saddle points.

In Hewett et al. (2019) a new technique was described for the construction of integral solutions to the *Parabolic Wave Equation*, typically with coalescing saddle points. Plots of some solutions were provided using `cuspint` (described below) in the cases that were "not too difficult", but others were excluded, for example, $A_{32}$ of equation (32) therein. This omission can now be easily produced using `PathFinder`, as shown in Figure 3.

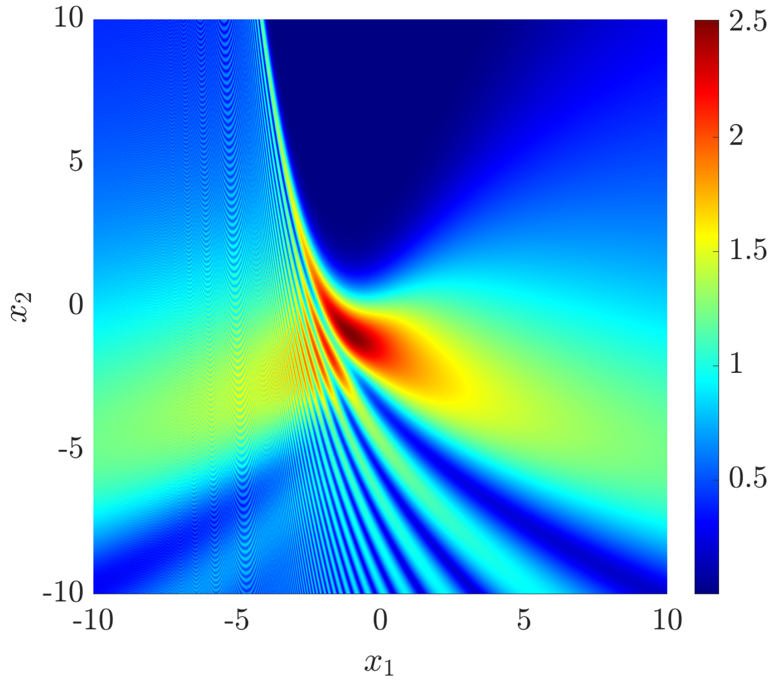


**Figure 3:** PathFinder approximation to $|A_{32}(x_1, x_2)|$, (32) of Hewett et al. (2019).

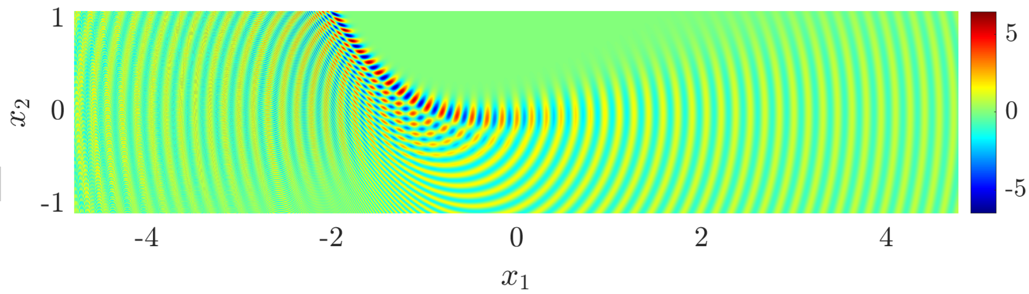


**Figure 4:** PathFinder approximation of a wavefield with a caustic near an inflection point, wavenumber 40.

The ideas of Hewett et al. (2019) were combined with `PathFinder` in Ockendon et al. (2024) and applied to the famous (unsolved) inflection point problem of Popov (1979). Via a simple change of variables, these solutions to the Parabolic Wave Equation could be transformed into meaningful solutions of the Helmholtz equation. Here `PathFinder` was used to visualise a wavefield with caustic behaviour close to a curve with an inflection point (as in Figure 4) and provided numerical validation of the asymptotic approximations therein.

## Comparison with other software

To the author's best knowledge, the only other software packages that can efficiently evaluate oscillatory integrals are Mathematica's `NIntegrate` function, when used with the `LevinRule` option (Wolfram, 2024), and the Fortan `cuspint` package (Kirk et al., 2000). We now briefly compare these packages against `PathFinder`.

An advantage of Mathematica's `NIntegrate` is that the oscillatory component does not always need to be factored explicitly; Mathematica does this symbolically. However, there are three drawbacks when compared to `PathFinder`:

- Based on experiments (Gibbs et al., 2024, sec. 5.3), `NIntegrate` does not appear to have a frequency-independent cost for general polynomial phase functions.
- `NIntegrate` does not work in general for an unbounded contour with complex endpoints.
- `NIntegrate` is not open source; the code cannot be seen or modified, and one must acquire a license to use it.

The `cuspint` package is similar to `PathFinder` in that it is also based on steepest descent contour deformation. There are two drawbacks when compared with `PathFinder`:

- The problem class is restricted to (1) when $(a, b) = \mathbb{R}$. Therefore, it may be used to model the catastrophe integrals of Figures 1 and 2, but not those of Figure 3 and 4.
- `cuspint` can experience "violent" exponential growth (Kirk et al., 2000, sec. 2), which can lead to inaccurate results. This is because, unlike `PathFinder`, it does not attempt a highly accurate approximation of the steepest descent contours.

In summary, `PathFinder` is the only existing software package that can be applied in general to (1).

# Acknowledgments

I am very grateful for the guidance of Daan Huybrechs and David Hewett throughout the development of this software. I am also grateful for financial support from KU Leuven project C14/15/05 and EPSRC projects EP/S01375X/1, EP/V053868/1.

Some of the code in `PathFinder` is copied from other projects. I acknowledge Aryo (2024), used for the Dijkstra shortest path algorithm. I also acknowledge Dirk Laurie and Walter Gautschi for writing the code used for the Golub-Welsch algorithm.

# References

Arnol'd, V. I. (1981). Lagrangian manifolds with singularities, asymptotic rays and the unfurled swallowtail. *Funktsional. Anal. I Prilozhen.*, *15*(4), 1–14, 96.

Aryo, D. (2024). *Dijkstra algorithm*. https://www.mathworks.com/matlabcentral/fileexchange/36140-dijkstra-algorithm

Deaño, A., Huybrechs, D., & Iserles, A. (2018). *Computing Highly Oscillatory Integrals*. SIAM.

Gibbs, A., Hewett, D. P., & Huybrechs, D. (2024). Numerical evaluation of oscillatory integrals via automated steepest descent contour deformation. *Journal of Computational Physics*, 112787. https://doi.org/10.1016/j.jcp.2024.112787

Hewett, D. P., Ockendon, J. R., & Smyshlyaev, V. P. (2019). Contour integral solutions of the parabolic wave equation. *Wave Motion*, *84*, 90–109. https://doi.org/10.1016/j.wavemoti.2018.09.015

Huybrechs, D., K., A., & Lejon, N. (2019). A numerical method for oscillatory integrals with coalescing saddle points. *SIAM J. Numer. Anal.*, *57*(6), 2707–2729. https://doi.org/10.1137/18M1221138

Huybrechs, D., & Vandewalle, S. (2006). On the evaluation of highly oscillatory integrals by analytic continuation. *SIAM J. Numer. Anal.*, *44*(3), 1026–1048. https://doi.org/10.1137/050636814

Kirk, N. P., Connor, J. N. L., & Hobbs, C. A. (2000). An adaptive contour code for the numerical evaluation of the oscillatory cuspoid canonical integrals and their derivatives. *Comp. Phys. Comm.*, *132*(1), 142–165. https://doi.org/10.1016/S0010-4655(00)00126-0

*NIST Digital Library of Mathematical Functions*. (2023). http://dlmf.nist.gov

Ockendon, J. R., Ockendon, H., Tew, R. H., Hewett, D. P., & Gibbs, A. (2024). A caustic terminating at an inflection point. *Wave Motion*, *125*, Paper No. 103257. https://doi.org/10.1016/j.wavemoti.2023.103257

Pearcey, T. (1946). The structure of an electromagnetic field in the neighbourhood of a cusp of a caustic. *Philos. Mag. (7)*, *37*, 311–317.

Popov, M. M. (1979). The problem of whispering gallery waves in a neighbourhood of a simple zero of the effective curvature of the boundary. *J. Sov. Math. (Now J. Math. Sci.)*, *11*, 791–797.

Wolfram. (2024). *Mathematica NIntegrate integration rules - LevinRule*. https://reference.wolfram.com/language/tutorial/NIntegrateIntegrationRules.html#32844337