

# <sup>1</sup> ODINN.jl: Scientific machine learning glacier modelling

<sup>3</sup> Jordi Bolibar  <sup>1,2</sup>¶, Facundo Sapienza  <sup>3,4</sup>, Alban Gossard<sup>1</sup>, Mathieu le Séac'h<sup>1</sup>, Vivek Gajadhar<sup>2</sup>, Fabien Maussion<sup>5,6</sup>, Bert Wouters<sup>2</sup>, and Fernando Pérez 

<sup>6</sup> 1 Univ. Grenoble Alpes, CNRS, IRD, G-INP, Institut des Géosciences de l'Environnement, Grenoble,  
<sup>7</sup> Franc <sup>2</sup> Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, The  
<sup>8</sup> Netherlands <sup>3</sup> Department of Geophysics, Stanford University, Stanford, United States <sup>4</sup> Department of  
<sup>9</sup> Statistics, University of California, Berkeley, United States <sup>5</sup> Bristol Glaciology Centre, School of  
<sup>10</sup> Geographical Sciences, University of Bristol, Bristol, UK <sup>6</sup> Department of Atmospheric and Cryospheric  
<sup>11</sup> Sciences, University of Innsbruck, Innsbruck, Austria ¶ Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

## Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Open Journals](#) ↗

## Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#))<sup>6</sup>

## <sup>12</sup> Summary

<sup>13</sup> ODINN.jl is a glacier model leveraging scientific machine learning (SciML) methods to perform  
<sup>14</sup> forward and inverse simulations of large-scale glacier evolution. It can simulate both surface  
<sup>15</sup> mass balance and ice flow dynamics through a modular architecture which enables the user  
<sup>16</sup> to easily modify model components. For this, ODINN.jl is in fact an ecosystem composed of  
<sup>17</sup> multiple packages, each one handling a specific task:

- Sleipnir.jl: Handles all the basic types, functions and datasets, common through the whole ecosystem, as well as data management tasks.
- Muninn.jl: Handles surface mass balance processes, via different types of models.
- Huginn.jl: Handles ice flow dynamics, by solving the ice flow partial differential equations (PDEs) using numerical methods. It can accommodate multiple types of ice flow models.
- ODINN.jl: Acts as the interface to the whole ecosystem, and provides the necessary tools to differentiate and optimize any model component. It can be seen as the SciML layer, enabling different types of inverse methods, using hybrid models combining differential equations with data-driven models.

<sup>27</sup> The ODINN ecosystem extends beyond this suite of Julia ([Bezanson et al., 2017](#)) packages, by leveraging the data preprocessing tools of the Open Global Glacier Model (OGGM, Maussion et al. ([2019](#))). We do so via an auxiliary Python library named Gungnir, which is responsible for downloading all the necessary data to force and initialize the model, such as glacier outlines from the Randolph Glacier Inventory (RGI Consortium ([2023](#)), RGI), digital elevation models (DEMs), ice thickness observations from GlaThiDa ([Consortium, 2020](#)), ice surface velocities from different studies ([Millan et al., 2022](#)), and many different sources of climate reanalyses and projections ([Eyring et al., 2016; Lange, 2019](#)). This implies that ODINN.jl, like OGGM, is virtually capable of simulating any of the ~274,000 glaciers on Earth ([RGI Consortium, 2023](#)).

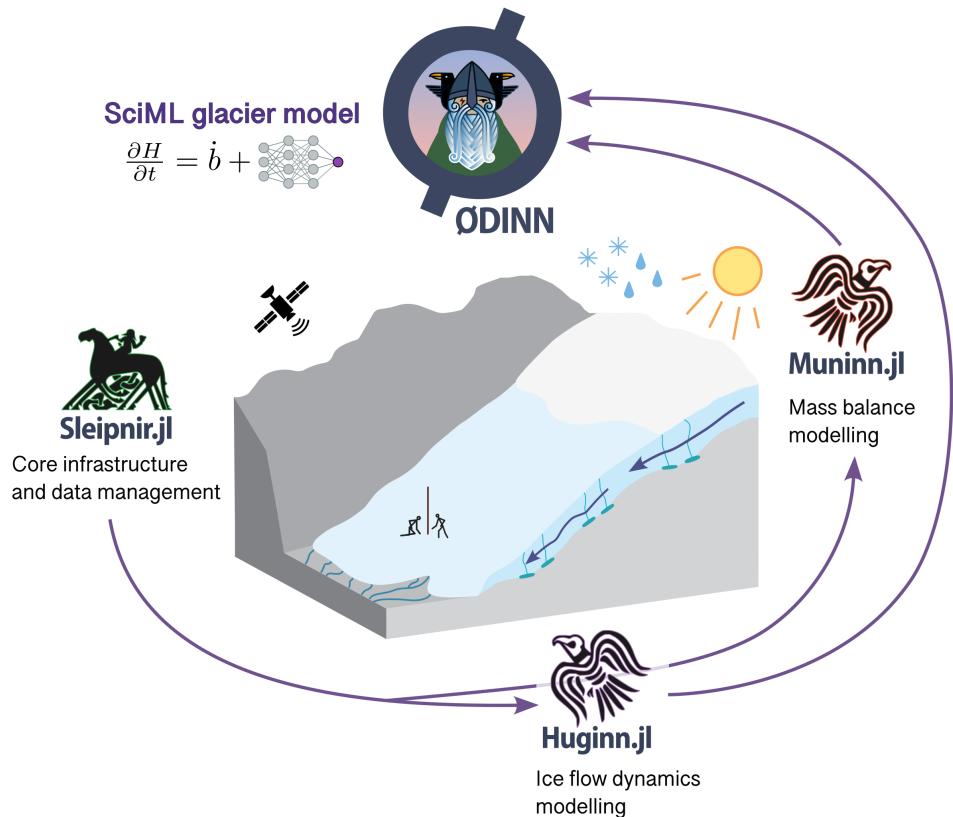


Figure 1: Overview of the ODINN.jl ecosystem.

36 ODINN.jl provides a high-level user-friendly interface, enabling the user to swap and replace  
 37 most elements of a glacier simulation in a very modular fashion. The main elements of a  
 38 simulation, such as the Parameters, a Model and a Simulation (either a Prediction or an  
 39 Inversion), are all objects that can be easily modified and combined. In a few lines of  
 40 code, the user can automatically retrieve all necessary information for most glaciers on Earth,  
 41 compose a Model based on a specific combination of surface mass balance and ice flow models,  
 42 and incorporate data-driven models (e.g. a neural network) to parametrize specific physical  
 43 processes of any of these components. Both forward and inverse simulations run in parallel  
 44 using multiprocessing, leveraging Julia's speed and performance. GPU compatibility is still not  
 45 ready, due to the difficulties of making everything compatible with automatic differentiation  
 46 (AD). Nonetheless, it is planned for future versions.

47 The most unique aspect of ODINN.jl is its differentiability and capabilities of performing all  
 48 sorts of different hybrid modelling. Since the whole ecosystem is differentiable, we can optimize  
 49 almost any model component, providing an extremely powerful framework to tackle many  
 50 scientific problems (Bolibar et al., 2023). ODINN.jl can optimize, separately or together, in a  
 51 steady-state or transient way:

- 52   ■ The initial or intermediate state of glaciers (i.e. their ice thickness) or the equivalent ice  
 53   surface velocities.
- 54   ■ Model parameters (e.g. the Glen coefficient A related to ice viscosity in a 2D Shallow  
 55   Ice Approximation (Hutter, 1983)), in a gridded or scalar format. This can be done for  
 56   multiple time steps where observations (e.g. ice surface velocities) are available.
- 57   ■ The parameters of a regressor (e.g. a neural network), used to parametrize a subpart  
 58   or one or more coefficients of an ice flow or surface mass balance mechanistic model.

59 This enables the exploration of empirical laws describing physical processes of glaciers,  
60 leveraging Universal Differential Equations (UDEs, Christopher Rackauckas et al. (2021)).

61 For this, it is necessary to differentiate (that is, computing gradients or derivatives) through  
62 complex code, including numerical solvers, which is a non-trivial task (Sapienza et al., 2024).  
63 We use reverse differentiation based on the adjoint method to achieve this. We have two  
64 strategies for computing both the adjoint and the required vector-jacobian products (VJPs):  
65 (1) manual adjoints, which have been implemented using AD via Enzyme.jl (Moses et al.,  
66 2021), as well as fully manual implementations of the discrete and continuous adjoints; and  
67 (2) automatic adjoints using SciMLSensitivity.jl (Chris Rackauckas et al., 2019), providing  
68 both continuous and discrete versions and available with different AD back-ends. These two  
69 approaches are complementary, with the manual adjoints being ideal for high-performance  
70 tasks, and serving as a ground truth for benchmarking and testing automatic adjoint methods  
71 from SciMLSensitivity.jl.

72 Beyond all these inverse modelling capabilities, ODINN.jl can also act as a more conventional  
73 forward glacier model, simulating glaciers in parallel, and easily customizing almost every  
74 possible detail of the simulation. Its high modularity, combined with the easy access to a vast  
75 array of datasets coming from OGGM, makes it very easy to run simulations, even with a  
76 simple laptop. Huginn.jl is responsible for the ice flow dynamics models, with an architecture  
77 capable of integrating and easily swapping various models. Models based on partial differential  
78 equations (PDEs) are solved using DifferentialEquations.jl (Christopher Rackauckas &  
79 Nie, 2017), which provides access to a huge amount of numerical solvers. For now, we have  
80 implemented a 2D Shallow Ice Approximation (SIA, Hutter (1983)), but in the future we plan to  
81 incorporate other models, such as the Shallow Shelf Approximation (SSA, Weis et al. (1999)).  
82 Validation of numerical forward simulations are evaluated in the test suite based on exact  
83 analytical solutions of the SIA equation for some simpler cases (Bueler et al., 2005). In terms  
84 of surface mass balance, Muninn.jl incorporates for now simple temperature-index models.  
85 Nonetheless, the main addition of the upcoming version will be the machine learning-based  
86 models from the MassBalanceMachine (Sjursen et al., 2025), which will become the de-facto  
87 solution. Frontal ablation (i.e. calving) and debris cover are not available for now, but we plan  
88 to add it to future versions of the model.

## 89 Statement of need

90 ODINN.jl has been designed to address the need for a glacier model which can leverage both the  
91 interpretability and established knowledge coming from the literature in the form of mechanistic  
92 models based on differential equations, with the flexibility and data-assimilation capabilities of  
93 data-driven models (Bolibar et al., 2023). The combination of these two paradigms enables a  
94 targeted approach to inverse methods for learning parametrizations of glacier physical processes,  
95 learning only the unknown physics and keeping a reliable structure in the dynamics in the form  
96 of a differential equation. While purely mechanistic and data-driven modelling approaches exist  
97 in glaciology, there is a need for flexible models which can leverage existing widely available  
98 observations at the glacier surface, to simulate complex physical processes of glaciers, such as  
99 basal sliding, creep or calving. Existing laws do not necessarily map available observations with  
100 these physical processes, difficulting the finding and calibration of parametrizations and laws.  
101 Approaches based on functional inversions and differentiable programming offer the needed  
102 flexibility to derive new empirical laws based on carefully chosen input proxies, which can help  
103 to test hypothesis of what can constitute and drive new parametrizations.

104 At the same time, a good representation of this complex and poorly represented physical  
105 processes is key to accurate predictions of glacier evolution, crucial for their impact to both  
106 freshwater resources and sea-level rise. Therefore, with ODINN.jl, we provide a unified modelling  
107 ecosystem, capable of both flexible and advance inverse methods for model calibration, as well  
108 as efficient and modular methods for forward simulations for large-scale glacier modelling.

## <sup>109</sup> Acknowledgements

<sup>110</sup> We acknowledge the help of Chris Rackauckas for the debugging and discussion of issues related  
<sup>111</sup> to the SciML Julia ecosystem, Redouane Lguensat for scientific discussions on the first prototype  
<sup>112</sup> of the model, and Julien le Sommer for scientific discussions around differentiable programming.  
<sup>113</sup> JB acknowledges financial support from the Nederlandse Organisatie voor Wetenschappelijk  
<sup>114</sup> Onderzoek, Stichting voor de Technische Wetenschappen (Vidi grant 016.Vidi.171.063) and a  
<sup>115</sup> TU Delft Climate Action grant. FS acknowledges funding from the National Science Foundation  
<sup>116</sup> (EarthCube programme under awards 1928406 and 1928374). AG acknowledges funding from  
<sup>117</sup> the MIAI cluster and Agence Nationale de la Recherche (ANR) in the context of France 2030  
<sup>118</sup> (grant ANR-23-IACL-0006).

## <sup>119</sup> References

- <sup>120</sup> Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A Fresh Approach to  
<sup>121</sup> Numerical Computing. *SIAM Review*, 59(1), 65–98. <https://doi.org/10.1137/141000671>
- <sup>122</sup> Bolibar, J., Sapienza, F., Maussion, F., Lguensat, R., Wouters, B., & Pérez, F. (2023). Universal  
<sup>123</sup> differential equations for glacier ice flow modelling. *Geoscientific Model Development*,  
<sup>124</sup> 16(22), 6671–6687. <https://doi.org/10.5194/gmd-16-6671-2023>
- <sup>125</sup> Bueler, E., Lingle, C. S., Kallen-Brown, J. A., Covey, D. N., & Bowman, L. N. (2005).  
<sup>126</sup> Exact solutions and verification of numerical models for isothermal ice sheets. *Journal of  
127 Glaciology*, 51(173), 291–306. <https://doi.org/10.3189/172756505781829449>
- <sup>128</sup> Consortium, G. (2020). *Glacier Thickness Database 3.1.0*. World Glacier Monitoring Service,  
<sup>129</sup> Zurich, Switzerland.
- <sup>130</sup> Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., & Taylor, K.  
<sup>131</sup> E. (2016). Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6)  
<sup>132</sup> experimental design and organization. *Geoscientific Model Development*, 9(5), 1937–1958.  
<sup>133</sup> <https://doi.org/10.5194/gmd-9-1937-2016>
- <sup>134</sup> Hutter, K. (1983). *Theoretical Glaciology*. Springer Netherlands. <https://doi.org/10.1007/978-94-015-1167-4>
- <sup>136</sup> Lange, S. (2019). *WFDE5 over land merged with ERA5 over the ocean (W5E5)*. GFZ Data  
<sup>137</sup> Services. <https://doi.org/10.5880/PIK.2019.023>
- <sup>138</sup> Maussion, F., Butenko, A., Champollion, N., Dusch, M., Eis, J., Fourteau, K., Gregor, P.,  
<sup>139</sup> Jarosch, A. H., Landmann, J., Oesterle, F., Recinos, B., Rothenpieler, T., Vlug, A., Wild, C.  
<sup>140</sup> T., & Marzeion, B. (2019). The Open Global Glacier Model (OGGM) v1.1. *Geoscientific  
141 Model Development*, 12(3), 909–931. <https://doi.org/10.5194/gmd-12-909-2019>
- <sup>142</sup> Millan, R., Mouginot, J., Rabatel, A., & Morlighem, M. (2022). Ice velocity and thickness  
<sup>143</sup> of the world's glaciers. *Nature Geoscience*, 15(2), 124–129. <https://doi.org/10.1038/s41561-021-00885-z>
- <sup>145</sup> Moses, W. S., Churavy, V., Paehler, L., Hückelheim, J., Narayanan, S. H. K., Schanen, M.,  
<sup>146</sup> & Doerfert, J. (2021). Reverse-mode automatic differentiation and optimization of GPU  
<sup>147</sup> kernels via enzyme. *Proceedings of the International Conference for High Performance  
148 Computing, Networking, Storage and Analysis*. <https://doi.org/10.1145/3458817.3476165>
- <sup>149</sup> Rackauckas, Chris, Innes, M., Ma, Y., Bettencourt, J., White, L., & Dixit, V. (2019).  
<sup>150</sup> DiffEqFlux.jl - A Julia Library for Neural Differential Equations. *arXiv:1902.02376 [Cs,  
151 Stat]*. <http://arxiv.org/abs/1902.02376>
- <sup>152</sup> Rackauckas, Christopher, Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., Skinner,  
<sup>153</sup> D., Ramadhan, A., & Edelman, A. (2021). *Universal Differential Equations for Scientific*

- 154 Machine Learning. arXiv. <https://doi.org/10.48550/arXiv.2001.04385>

155 Rackauckas, Christopher, & Nie, Q. (2017). DifferentialEquations.jl – A Performant and  
156 Feature-Rich Ecosystem for Solving Differential Equations in Julia. *Journal of Open*  
157 *Research Software*, 5, 15. <https://doi.org/10.5334/jors.151>

158 RGI Consortium. (2023). *Randolph Glacier Inventory - A Dataset of Global Glacier Outlines*,  
159 Version 7. National Snow; Ice Data Center. <https://doi.org/10.5067/F6JMOVY5NAVZ>

160 Sapienza, F., Bolibar, J., Schäfer, F., Groenke, B., Pal, A., Boussange, V., Heimbach,  
161 P., Hooker, G., Pérez, F., Persson, P.-O., & Rackauckas, C. (2024). *Differentiable*  
162 *Programming for Differential Equations: A Review*. arXiv. <http://arxiv.org/abs/2406.09699>

163

164 Sjursen, K. H., Bolibar, J., Meer, M. van der, Andreassen, L. M., Biesheuvel, J. P., Dunse,  
165 T., Huss, M., Maussion, F., Rounce, D. R., & Tober, B. (2025). Machine learning  
166 improves seasonal mass balance prediction for unmonitored glaciers. *EGUsphere*, 1–39.  
167 <https://doi.org/10.5194/egusphere-2025-1206>

168 Weis, M., Greve, R., & Hutter, K. (1999). Theory of shallow ice shelves. *Continuum Mechanics*  
169 and *Thermodynamics*, 11(1), 15–50. <https://doi.org/10.1007/s001610050102>

Bolibar et al. (2025). ODINN.jl: Scientific machine learning glacier modelling. *Journal of Open Source Software*, VOL?(ISSUE?), PAGE? 5  
<https://doi.org/10.xxxxxx/draft>.