Supplementary Information: Factorized Fourier Neural Operator Surrogate for Basin-Scale Tsunami Propagation in the East Sea

Jinyoung Kim[1], Myung Jin Koh[1], Seung-taek Oh[1], and Sangyoung Son[1]

[1]School of Civil, Environmental and Architectural Engineering, Korea University, Seoul 02841, South Korea

Correspondence: Sangyoung Son (sson@korea.ac.kr), Myung Jin Koh (myungj@korea.ac.kr)

This supplemental material includes Figs. S1–S2, which are the velocity comparisons at the virtual buoy stations, and Tables S1–S3 that compare the baseline standard (non-factorized) FNO models with the F-FNO surrogate model investigated in the paper. Table S4 includes the COMCOT–PCOMCOT solver comparison to give an idea of the runtime reference mentioned in the paper.

S1 Velocity time series at virtual buoys

The main text presents free-surface elevation (η) time series at six monitoring buoys. In Figs. S1 and S2 the corresponding depth-averaged velocity components u and v are given for the same situation and buoys.

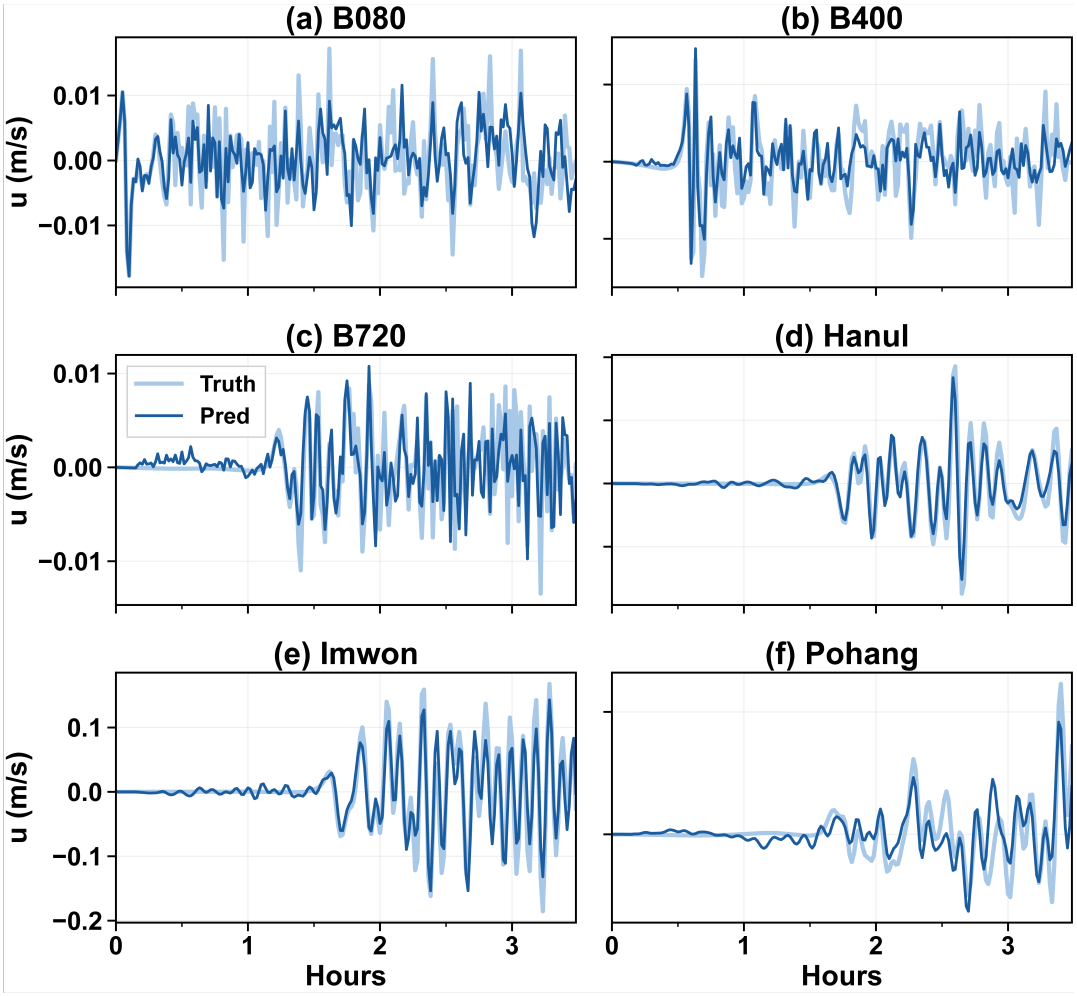

Figure S1. Depth-averaged eastward velocity (u) time series at six monitoring buoys for the same scenario shown in Fig.11 of the main text. Light shading: COMCOT reference (Truth); dark line: F-FNO prediction (Pred).

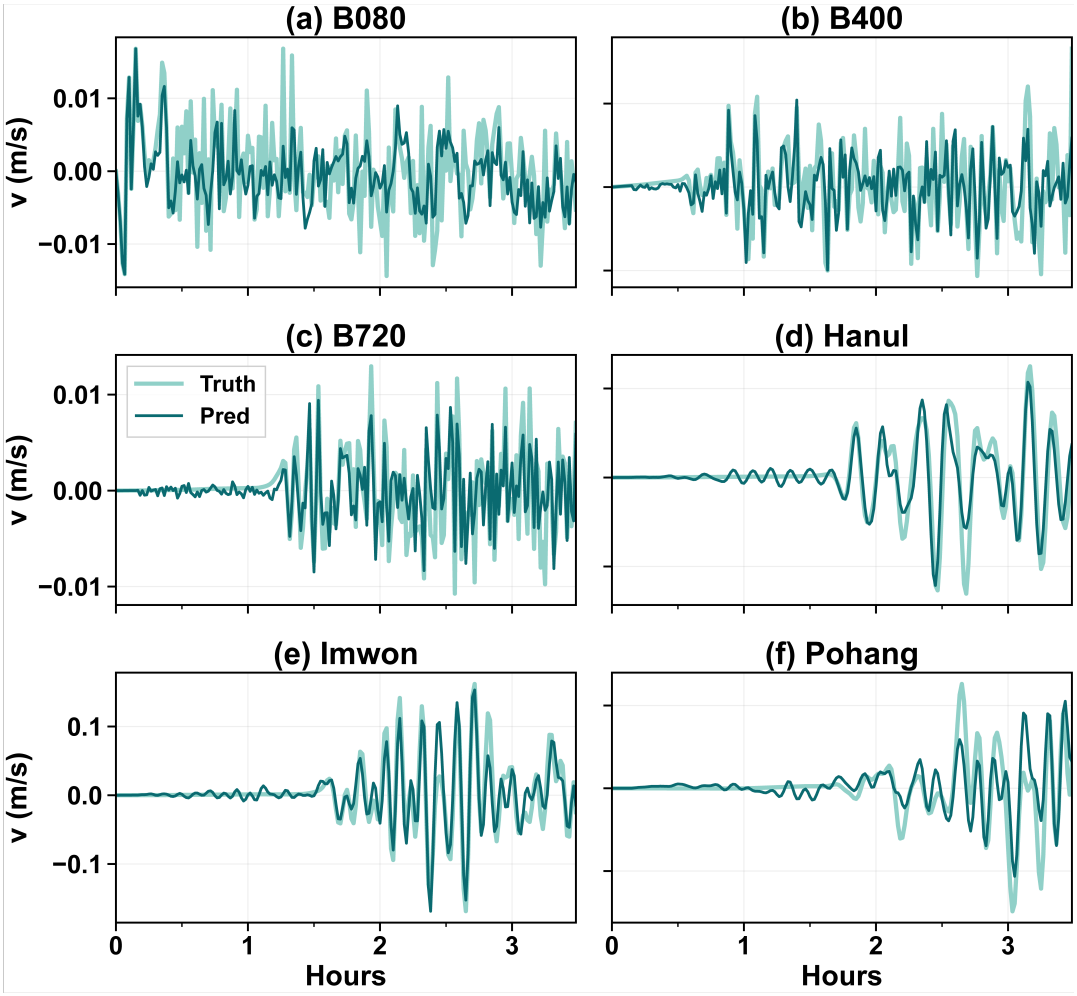

Figure S2. Depth-averaged northward velocity (v) time series at six monitoring buoys for the same scenario shown in Fig.11 of the main text. Light shading: COMCOT reference (Truth); dark line: F-FNO prediction (Pred).

S2 Standard FNO baseline comparison

For validation of the factorized spectral formulation, we train standard (non-factorized) FNO models on the COMCOT. The same FNO variants and configurations are trained and tested on the same held-out test set splits as before. The standard FNO stores a full 2-D spectral weight tensor $R \in \mathbb{C}^{M_x \times M_y \times d_v \times d_v}$ in each layer. So it can fit a smaller mode count on a single GPU (with $M{=}64$) as compared to the factorized FNO ($M{=}256$). We use the same training protocol, loss terms and data splits as with the F-FNO models in the main text, but with the configurations summarized in Table S1.

Table S1. Standard FNO model configurations. All variants use the same loss weights, training schedule, and data splits as the F-FNO experiments in the main text. Differences from the FNO-A baseline are underlined.

	FNO-A	**FNO-B**	**FNO-C**	**FNO-D**
Architecture	FNO	FNO	FNO	FNO
Width (d_v)	64	64	64	40
Depth (layers)	10	10	8	10
Modes (M_x, M_y)	(64,64)	(64,64)	(64,64)	(32,32)
Learning rate	3×10^{-4}	5×10^{-4}	3×10^{-4}	3×10^{-4}
Epochs	20	20	20	20

Table S2. Standard FNO held-out test performance (mean $\pm$ std over test scenarios). For each metric and split the best (lowest mean) value is in **bold**. ATE statistics are computed only over scenarios with a detected first arrival; N_{ATE} denotes the valid count.

Test split	Model	RMSE$_\eta$ (m)	ATE (min)	BEE	RMSE$_{\mathrm{avg}}$	N	N_{ATE}
Test-E (Ep1, M1–M3)	FNO-A	0.0433 ± 0.0193	29.2 ± 61.0	$\mathbf{0.2366 \pm 0.0300}$	0.0198 ± 0.0093	162	51
	FNO-B	0.0432 ± 0.0192	$\mathbf{8.8 \pm 34.3}$	0.2618 ± 0.0399	$\mathbf{0.0198 \pm 0.0093}$	162	43
	FNO-C	$\mathbf{0.0432 \pm 0.0192}$	11.9 ± 36.4	0.2481 ± 0.0231	0.0198 ± 0.0093	162	38
	FNO-D	0.0435 ± 0.0193	42.5 ± 70.8	0.3430 ± 0.0177	0.0199 ± 0.0093	162	60
Test-M (Ep2–Ep4, M4)	FNO-A	0.0817 ± 0.0178	13.6 ± 13.7	0.1264 ± 0.0185	0.0386 ± 0.0105	162	151
	FNO-B	$\mathbf{0.0779 \pm 0.0176}$	15.0 ± 18.2	$\mathbf{0.0898 \pm 0.0157}$	$\mathbf{0.0370 \pm 0.0103}$	162	136
	FNO-C	0.0835 ± 0.0192	$\mathbf{8.7 \pm 8.8}$	0.1574 ± 0.0214	0.0397 ± 0.0112	162	137
	FNO-D	0.0987 ± 0.0234	27.3 ± 28.0	0.3259 ± 0.0329	0.0471 ± 0.0131	162	109
Test-EM (Ep1, M4)	FNO-A	0.1009 ± 0.0242	16.3 ± 20.2	0.1413 ± 0.0294	0.0501 ± 0.0125	54	51
	FNO-B	$\mathbf{0.0995 \pm 0.0247}$	10.4 ± 10.2	$\mathbf{0.1223 \pm 0.0436}$	$\mathbf{0.0495 \pm 0.0127}$	54	50
	FNO-C	0.1008 ± 0.0242	14.1 ± 19.6	0.1636 ± 0.0258	0.0501 ± 0.0124	54	49
	FNO-D	0.1007 ± 0.0250	$\mathbf{1.7 \pm 5.7}$	0.3101 ± 0.0178	0.0502 ± 0.0128	54	46
Overall (all held-out)	FNO-A	0.0680 ± 0.0194	17.3 ± 24.5	0.1758 ± 0.0250	0.0322 ± 0.0103	378	253
	FNO-B	$\mathbf{0.0661 \pm 0.0193}$	12.9 ± 19.5	$\mathbf{0.1681 \pm 0.0300}$	$\mathbf{0.0314 \pm 0.0102}$	378	229
	FNO-C	0.0687 ± 0.0199	$\mathbf{10.4 \pm 15.9}$	0.1971 ± 0.0227	0.0326 ± 0.0105	378	224
	FNO-D	0.0753 ± 0.0219	26.1 ± 35.2	0.3310 ± 0.0243	0.0359 ± 0.0115	378	215

Table S3. Head-to-head comparison of the best F-FNO (Selected) and the best standard FNO per split (chosen by lowest mean RMSE_η). Δ: relative change (negative = F-FNO is better). Values are means over test scenarios; standard deviations are in Tables S2 and 3 of the main text.

Split	Model	RMSE_η (m)	ATE (min)	BEE	$\mathrm{RMSE}_{\mathrm{avg}}$
Test-E	F-FNO Selected	0.028	7.5	0.039	0.013
	FNO-C	0.043	11.9	0.248	0.020
	Δ	-36%	-37%	-84%	-37%
Test-M	F-FNO Selected	0.058	8.5	0.016	0.028
	FNO-B	0.078	15.0	0.090	0.037
	Δ	-26%	-43%	-82%	-25%
Test-EM	F-FNO Selected	0.076	12.1	0.031	0.038
	FNO-B	0.100	10.4	0.122	0.050
	Δ	-23%	$+16\%$	-75%	-23%
Overall	F-FNO Selected	0.048	8.9	0.028	0.023
	FNO-B	0.066	12.9	0.168	0.031
	Δ	-28%	-31%	-83%	-28%

See Table S3 for the comparison of our best-performing F-FNO model (Selected; 10 layers, d_v=64, M=256, factorized spectral layers) with the best-performing standard FNO model from those with the lowest mean RMSE_η on each OOD split. F-FNO models achieve 24–36 % reductions in RMSE_η for all three held-out splits and decreases BEE by 75–84 % when compared to the best-performing standard FNO model on each split. In Test-EM, the best standard FNO variant achieves a slightly lower mean ATE, but with much larger field-wise and spectral errors. This comparison is consistent with our pilot experiments. In long autoregressive prediction, the standard FNO often showed progressive weakening of the predicted free-surface elevation η, even after additional loss functions were introduced. After switching to the F-FNO with larger retained modes, we obtained more stable long-horizon predictions.

S3 F-FNO minimum water depth sensitivity

A validation sensitivity test with minimum wet-depth thresholds of 1, 2, 3, and 5 m showed that the 5 m setting gave the best overall validation performance (Table S4). We therefore used 5 m in the main experiments.

Table S4. Validation sensitivity to the minimum wet-depth threshold. All runs in this sweep used the same 8-GPU distributed data parallel (DDP) training setup, and only the minimum wet-depth threshold was changed. For each threshold, the checkpoint with the lowest validation weighted L1 loss is reported.

Depth threshold (m)	Best val(L1w)	val(MSE)	val RMSE_{η} (m)	val $\mathrm{RMSE}_{\mathrm{avg}}$ (m)
1.0	0.1244	0.9077	0.0965	0.0580
2.0	0.1131	0.4377	0.0843	0.0468
3.0	0.1119	0.3467	0.0839	0.0444
5.0	**0.1096**	**0.2130**	**0.0820**	**0.0403**

S4 Context for the PCOMCOT runtime reference

The entries in Table S5 describe the differences between the methods of calculation employed in COMCOT and PCOMCOT. The table is to clarify for users that times for PCOMCOT should not be compared directly with the times quoted in the main text. The governing equations and dispersion relationship are different. The arithmetic precision and the parallelization strategy also differ between the two models.

Table S5. A short summary of COMCOT vs. PCOMCOT that was used for the runtime calculations in the main text. PCOMCOT details follow Zhu et al., 2024

Feature	**COMCOT v1.7**	**PCOMCOT**
Governing equations	Linear shallow-water equations (continuity + two horizontal momentum equations)	Boussinesq-type dispersive formulation with non-hydrostatic correction, including an additional vertical-momentum/pressure-related solve
Dispersion treatment	No physical frequency dispersion in the governing equations	Dispersive wave effects represented through the non-hydrostatic / Boussinesq formulation
Time integration	Explicit leap-frog type update with fixed cost per time step	Semi-implicit procedure with an additional iterative linear solve; cost depends on convergence
Spatial discretization	Staggered-grid finite-difference scheme	Staggered-grid framework with additional flux-centered / non-hydrostatic treatment for dispersive terms
Arithmetic precision	Single precision (float32)	Double precision (float64)
Parallelization	Serial in the single-layer setup used here	MPI-based domain decomposition on multiple CPU cores
Primary use in this study	Reference solver used to generate the surrogate training and test database	External high-fidelity runtime reference for contextual comparison only