대한조선학회논문집 Research Paper
Journal of the Society of Naval Architects of Korea
eISSN:2287-7355, Vol. 59, No. 6, pp. 393-399, December 2022
https://doi.org/10.3744/SNAK.2022.59.6.393

그래프 신경망을 이용한 단순 선박 선형의 저항성능 시뮬레이션

박태원[1]·김인섭[2]·이훈[3]·박동우[4,†]
동명대학교 조선해양시뮬레이션센터[1]
한국해양교통안전공단 스마트안전연구실[2]
㈜토탈소프트뱅크 물류시스템연구소[3]
동명대학교 조선해양공학과[4]

Resistance Performance Simulation of Simple Ship Hull Using Graph Neural Network

TaeWon Park[1]·Inseob Kim[2]·Hoon Lee[3]·Dong-Woo Park[4,†]
Shipbuilding & Marine Simulation Center, Tongmyong University[1]
Smart Safety Research Department, Korea Maritime Transportation Safety Authority[2]
Logistics System Institute, Total Soft Bank, Ltd.[3]
School of Naval Architecture & Ocean Engineering, Tongmyong University[4]

During the ship hull design process, resistance performance estimation is generally calculated by simulation using computational fluid dynamics. Since such hull resistance performance simulation requires a lot of time and computation resources, the time taken for simulation is reduced by CPU clusters having more than tens of cores in order to complete the hull design within the required deadline of the ship owner. In this paper, we propose a method for estimating resistance performance of ship hull by simulation using a graph neural network. This method converts the 3D geometric information of the hull mesh and the physical quantity of the surface into a mathematical graph, and is implemented as a deep learning model that predicts the future simulation state from the input state. The method proposed in the resistance performance experiment of simple hull showed an average error of about 3.5 % throughout the simulation.

Keywords : Mesh based simulation(메시 기반 시뮬레이션), Graph neural network(그래프 신경망), Resistance performance(저항성능), Computational fluid dynamics(전산유체역학)

1. 서 론

선박 선형의 초기 설계는 선주의 요구 사항과 유사한 실적선을 선정하고 요구 사항에 맞게 주요치수를 조정한 다음, 유체성능을 추정하는 과정으로 진행된다. 이런 과정 중에 유체성능 추정은 주로 전산유체역학(Computational Fluid Dynamics, CFD)을 사용한 시뮬레이션을 사용하기 때문에 많은 시간과 연산자원이 필요하다. 따라서 현실적인 상황에서 수십 개 이상의 코어를 가진 워크스테이션이나 서버급 CPU를 통해 CFD 시뮬레이션에 필요한 시간을 줄이는 것이 일반적이다.

최근에는 물리학적인 시뮬레이션에 심층학습 기반의 인공신경망을 활용하는 연구사례가 늘고 있다. 전형적인 정규격자 기반의 시뮬레이션은 2차원 픽셀이나 3차원 복셀 공간을 주로 이용하며, 일반적으로 정규격자 입력에 대해서 뛰어난 성능을 보이는 합성곱 신경망으로 구현한다 (Kim et al., 2019; Thuerey et al., 2019). 하지만 정규격자 방식은 선체의 표면과 같은 3차원 곡면을 직접 표현하기 힘든 단점이 있으므로 비정규격자 방식의 인공신경망으로 이를 극복하려는 연구가 나타나고 있다. 그중에서 Pfaff et al. (2021)은 CFD에서 주로 사용하는 비정규격자 방식인 메시 기반의 시뮬레이션에 그래프 신경망(Graph Neural Network, GNN)을 활용하였다.

Received : 27 September 2022 | Revised : 18 October 2022 | Accepted : 1 November 2022
† Corresponding author : Dong-Woo Park, dwpark@tu.ac.kr

본 연구에서는 인공지능을 사용한 3차원 선형의 저항성능 시뮬레이션 방법을 소개한다. 시뮬레이션하려는 선형의 기하학적 정보와 표면의 물리량을 GNN에 입력하여 미래의 물리량을 추론하는 방법을 제안한다. 2장에서는 시뮬레이션에서 모델링하는 메시의 물리량과 기하학적 정보를 그래프로 표현하는 방법과 GNN의 작동원리를 소개한다. 3장에서는 실험에 사용한 시뮬레이션 데이터를 수집한 방법과 GNN의 세부 구조를 설명하며, 4장에서는 기존의 CFD를 사용한 방식과 제안하는 방식의 저항성능 시뮬레이션 결과를 비교해서 보인다.

2. GNN 모델

2.1 모델 개요

MeshGraphNets (Pfaff et al., 2021)는 메시 기반의 시뮬레이션을 위해 설계한 GNN 모델로, 메시의 점과 선을 그래프의 노드와 간선으로 변환할 수 있으므로 다차원 메시 데이터를 그래프로 직접 표현할 수 있다. 모델의 입력인 시뮬레이션 시간 단계 t에서 메시의 상태 M^t를 식 (1)과 같이 그래프로 정의한다.

$$M^t = (V, E) \tag{1}$$

여기서 V는 그래프의 노드 집합, E는 그래프의 간선 집합을 의미한다. 각 노드 $node_i \in V$는 참조 메시의 공간 좌표 위치 u_i의 점과 연관되며, 각 간선 $edge_{ij} \in E$는 $node_i$에서 $node_j$로 나가는 방향의 선과 연관된다. 시뮬레이션 상태 M^t가 주어지면 모델은 Fig. 3처럼 부호화(encoder), 처리(processor), 복호화(decoder) 과정을 거쳐 다음 시간 단계의 시뮬레이션 상태 M^{t+1}을 출력으로 예측한다.

모델의 부호화 과정에서 $node_i$는 모델링을 위한 물리량 q_i를, $edge_{ij}$는 u_i와 u_j 사이의 상대 변위 벡터 $u_{ij} = u_i - u_j$와 해당 벡터의 노름 $|u_{ij}|$를 잠재 벡터 v_i와 e_{ij}로 각각 부호화하며, 부호화에는 다층 퍼셉트론(multilayer perceptron, MLP)과 층 정규화(layer normalization, LN)를 사용한다. 처리 과정에서는 각 노드의 잠재 벡터 v_i로 부호화된 정보를 해당 노드와 간선으로 직접 연결된 주변 노드에 메시지로 전달한다.

$$e'_{ij} \leftarrow f^E(e_{ij}, v_i, v_j) \tag{2}$$

$$v'_i \leftarrow f^V(v_i, \sum_j e'_{ij}) \tag{3}$$

각 메시지 전달 층은 잠재 벡터 v_i와 e_{ij}를 식 (2)와 식 (3)의 방식으로 v'_i와 e'_{ij}로 갱신하며, 이를 위한 f^V와 f^E는 잔차 연결(residual connection)을 사용한 MLP로 구현한다. 처리 과정에서 여러 층에 걸쳐 메시지를 전달하므로 간선으로 직접 연결되어 있지 않은 노드라도 서로의 지역적인 정보를 간접적으로 교환할 수 있다. 복호화 과정에서는 시간 단계 t의 입력 상태로부터 시간 단계 t+1의 상태를 예측하기 위해서, 각 노드의 잠재 벡터 v_i를 물리량 q_i^t의 변화량인 p_i로 복호화한다.

$$q_i^{t+1} = q_i^t + p_i \tag{4}$$

식 (4)의 방식으로 모델은 시간 단계 t+1의 물리량 q_i^{t+1}을 추론하므로, 현재 시뮬레이션 상태 M^t를 통해서 한 단계의 미래 상태 M^{t+1}을 예측할 수 있다.

2.2 GNN을 이용한 시뮬레이션

한 단계의 미래 상태를 예측할 수 있는 모델은 식 (5)와 같은 방식으로 이전 시간 단계 상태 M^{t-1}을 통해서 예측한 상태 M^t를 다시 모델의 입력으로 하여, 그다음 시간 단계 상태인 M^{t+1}을 예측할 수 있다.

$$M^{t+1} \leftarrow f^M(M^t) \leftarrow f^M(f^M(M^{t-1})) \tag{5}$$

여기서 f^M은 그래프 신경망 모델을 의미하며 이러한 자기회귀(autoregressive) 방식으로 연속적인 시뮬레이션이 가능하므로 최초 상태 M^0가 모델의 입력으로 주어지면 시뮬레이션 상태의 시계열 $\{M^1, M^2, \ldots, M^n\}$을 예측할 수 있다.

3. 실 험

3.1 대상 및 데이터

실험을 위한 선박 선형은 Fig. 1에서 보이는 것처럼 단순한 선형에서 선미 부분에 단이 지는 형태로 설계하였다. 여러 경우에서 모델의 성능을 검증하기 위해 기본 크기의 선형에서 폭의 크기를 -9 %에서 +9 %까지 1 % 단위로 조절한 19개의 선형을 사용하였으며, 그중에 3개(-5 %, 0 %, +5 %)를 테스트용으로 정하고 나머지 16개는 학습용으로 정하여 실험을 진행하였다.

실험 대상 선형에 대한 저항성능 추정을 위해서 데이터집합은 CFD 시뮬레이션을 사용해서 수집하였다. 실험에 사용한 계산용 격자는 Fig. 2와 같으며, 선형의 제원과 CFD 시뮬레이션에 사용한 매개변수는 Table 1과 같다. 학습을 위한 시뮬레이션의 총 시

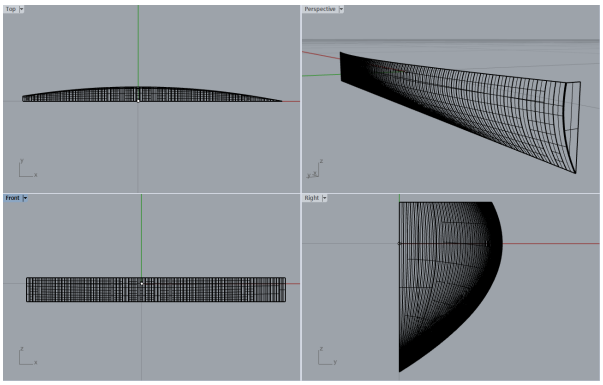


Fig. 1 Simple ship hull for experiment.

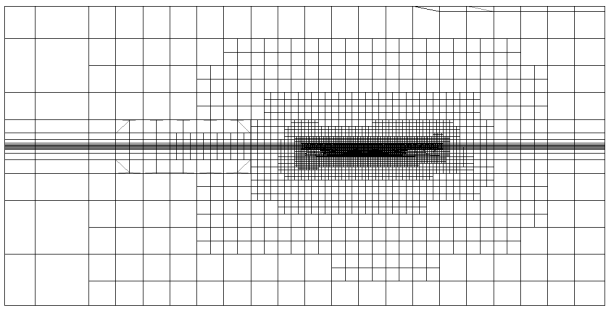


Fig. 2 Computational grid for experiment.

Table 1 Specifications of simple ship hull and parameters of CFD simulation.

Length	0.9 m
Breadth	0.16 m
Draft	0.063 m
Design speed	0.891 m/s
Simulaton time	5 s
△time	0.02 s
Number of steps	250
Number of grids	48258

간은 5초이며, 매 0.02초 시간 간격으로 한 선형당 250단계의 시뮬레이션 상태를 메시 형태로 수집하였다.

시뮬레이션에서 모델링할 물리량으로 저항성능 계산에 필요한 전압(total pressure)과 좌표축별 벽 전단응력(wall shear stress)을 사용하였으며, 물 부피분율(water volume fraction)을 부수적으로 사용하였다.

3.2 모델 구조 및 학습 방법

실험에 사용한 GNN 모델의 전체적인 구조는 Fig. 3과 같다. 모델은 크게 노드의 물리량과 간선의 기하학적 정보를 잠재 벡터로 변환하는 부호화 부분, 그래프의 노드가 간선을 통해 서로 정보를 교환하는 처리 부분, 노드의 잠재 벡터를 물리량의 변화량으로 추론하는 복호화 부분으로 나뉜다. 입출력 벡터 X의 각 성분 $x_i \in X$는 식 (6)과 같이 평균 μ_i와 표준편차 σ_i를 사용해 정규화하였다.

$$x'_i = \frac{x_i - \mu_i}{\sigma_i} \tag{6}$$

모델에서 복호화 과정을 제외한 모든 MLP의 출력값은 LN을 사용하도록 구성하여 심층학습 과정을 안정화시켰다 (Ba et al., 2016). 또한 신경망 학습 중에 다수의 메시지 전달 층을 쌓아 올린 모델의 그래디언트를 원활히 역전파하기 위해 메시지 전달 과정에서 잔차 연결을 사용하였다 (He et al., 2016). 부호화 과정과 처리 과정에서 2개 이상의 특징을 1개의 벡터로 모을 때는 접합(concatenate) 연산을 사용하였다.

신경망 모델의 연결강도 매개변수 최적화에는 Adam을 사용하였으며 (Kingma and Ba, 2015), 물리량의 예측 변화량과 실제 변화량의 차이를 줄이기 위한 손실함수는 평균제곱오차를 사용하였다. 모델 학습에 사용한 초매개변수는 Table 2와 같다.

학습은 총 500 에포크 동안 수행하였으며, 배치 크기는 8로 설정하였다. 최초 학습률은 1.0×10^{-4}로 설정하였고 빠른 수렴을 위해서 300번의 에포크 후에 학습률을 1.0×10^{-5}로 낮추었다. 복호화 부분을 제외한 모델에 포함된 모든 MLP는 길이가 128인 잠

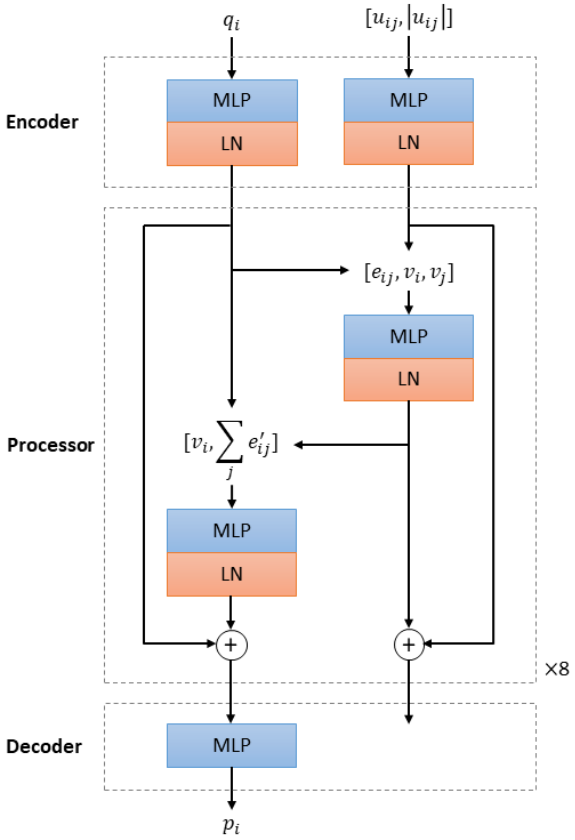


Fig. 3 Graph neural network architecture used in this paper (+: add, [·]: concatenate).

Table 2 Hyperparameters for graph neural network learning (MP*: message passing).

Number of epochs	500
Batch size	8
Optimizer	Adam
Initial learning rate	1.0×10^{-4}
Learning rate decay	1.0×10^{-5} (after 300th epoch)
Loss function	Mean squared error
Activation function	SiLU
Length of latent vector	128
Number of hidden layers	2
Number of MP* layers	8

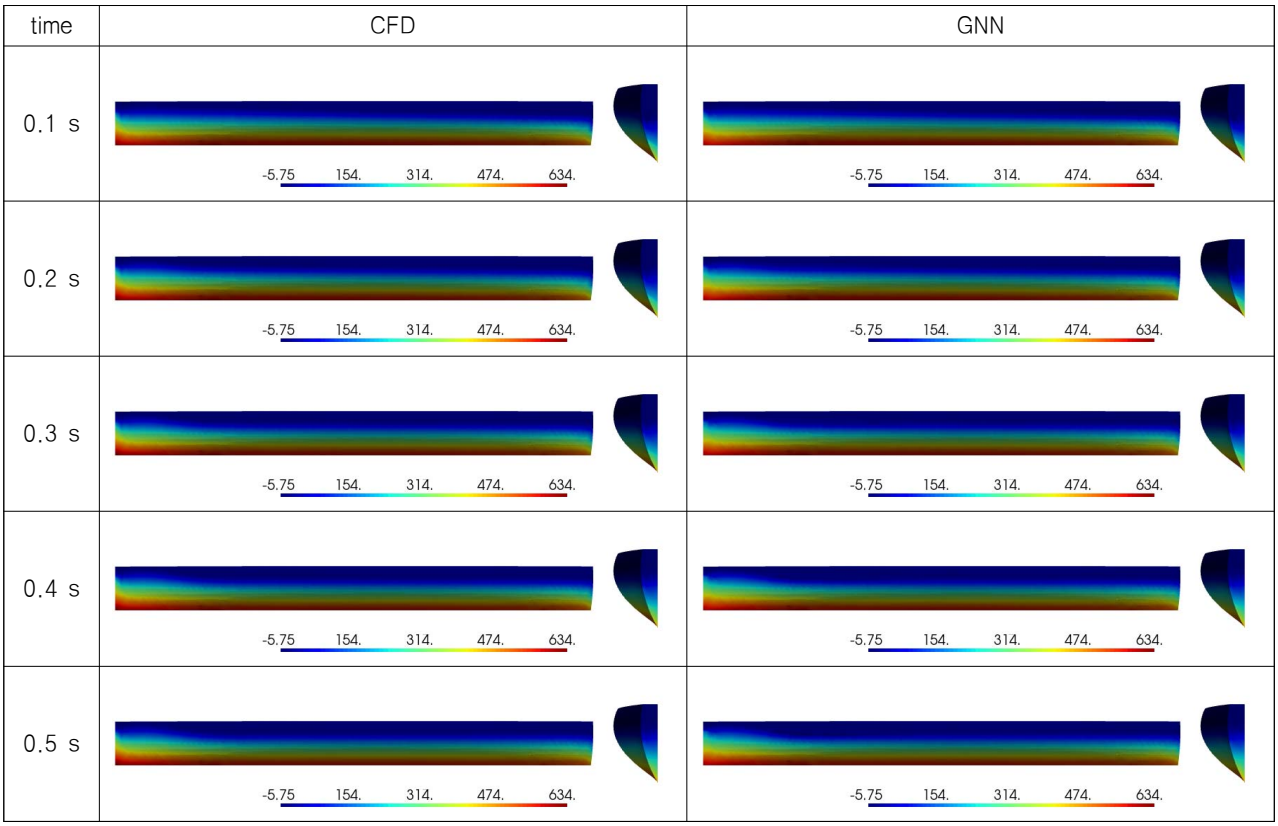


Fig. 4 Total pressure examples simulated with base breadth hull.

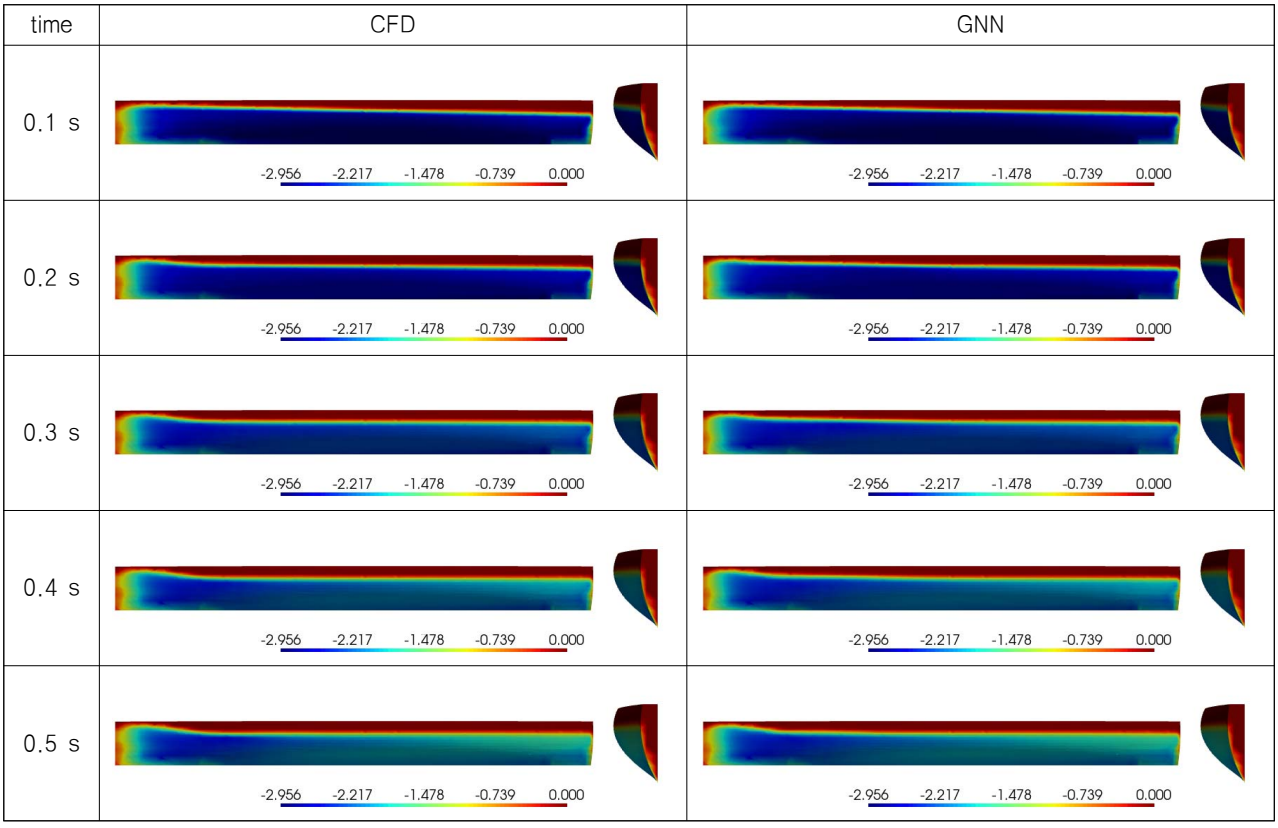


Fig. 5 Wall shear stress (x-axis) examples simulated with base breadth hull.

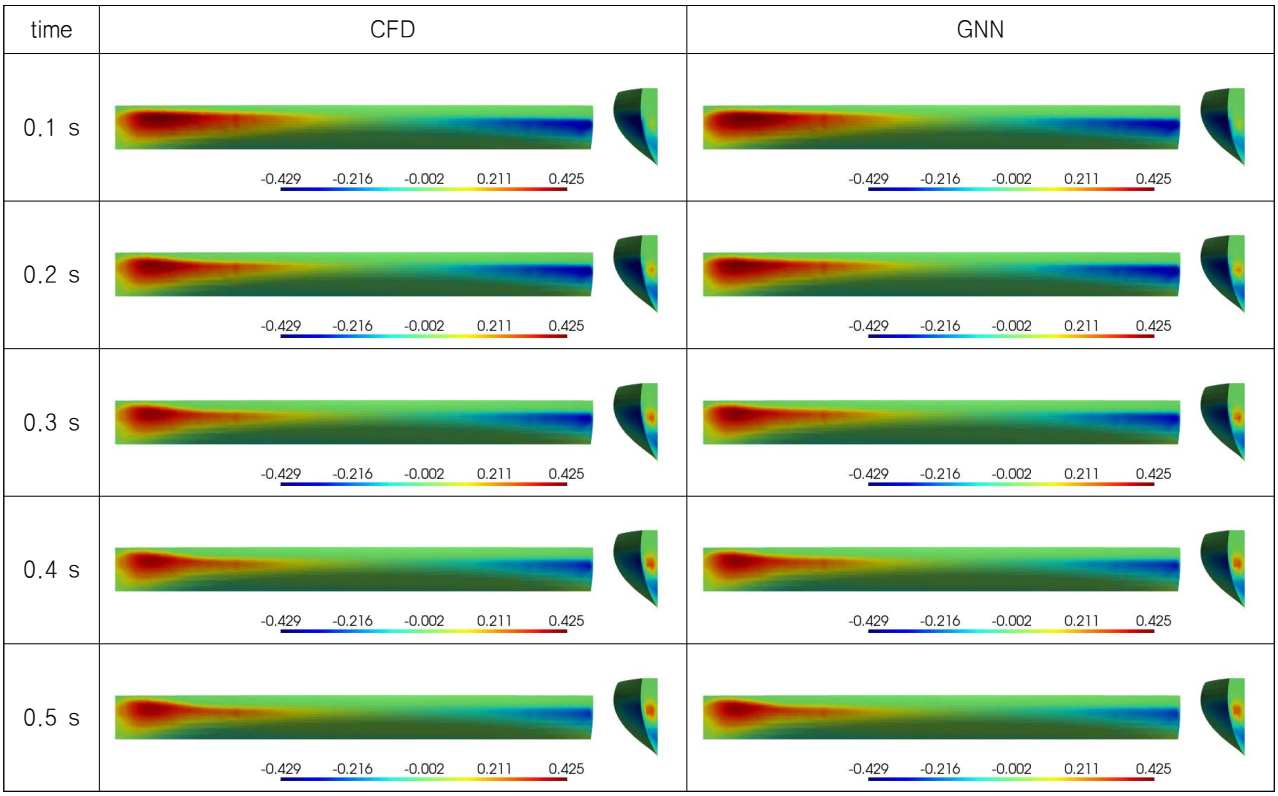


Fig. 6 Wall shear stress (y-axis) examples simulated with base breadth hull.

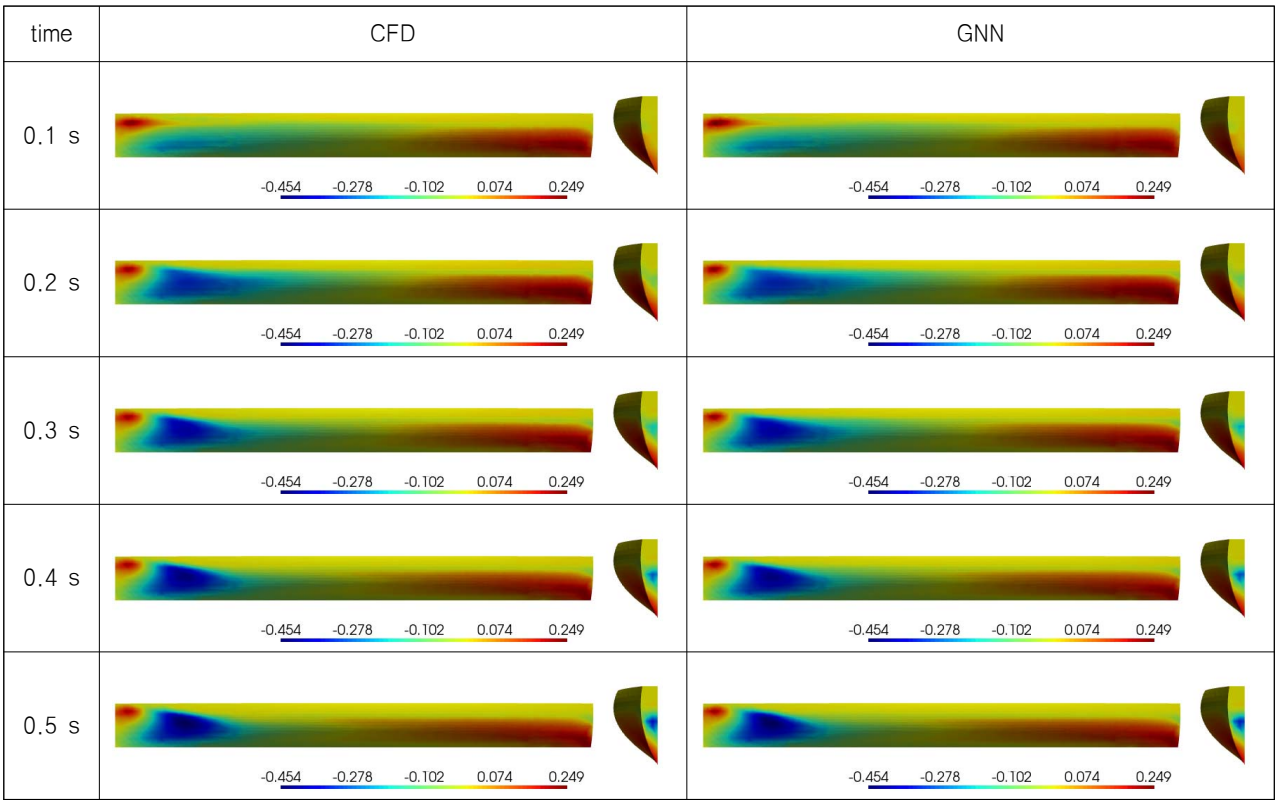


Fig. 7 Wall shear stress (z-axis) examples simulated with base breadth hull.

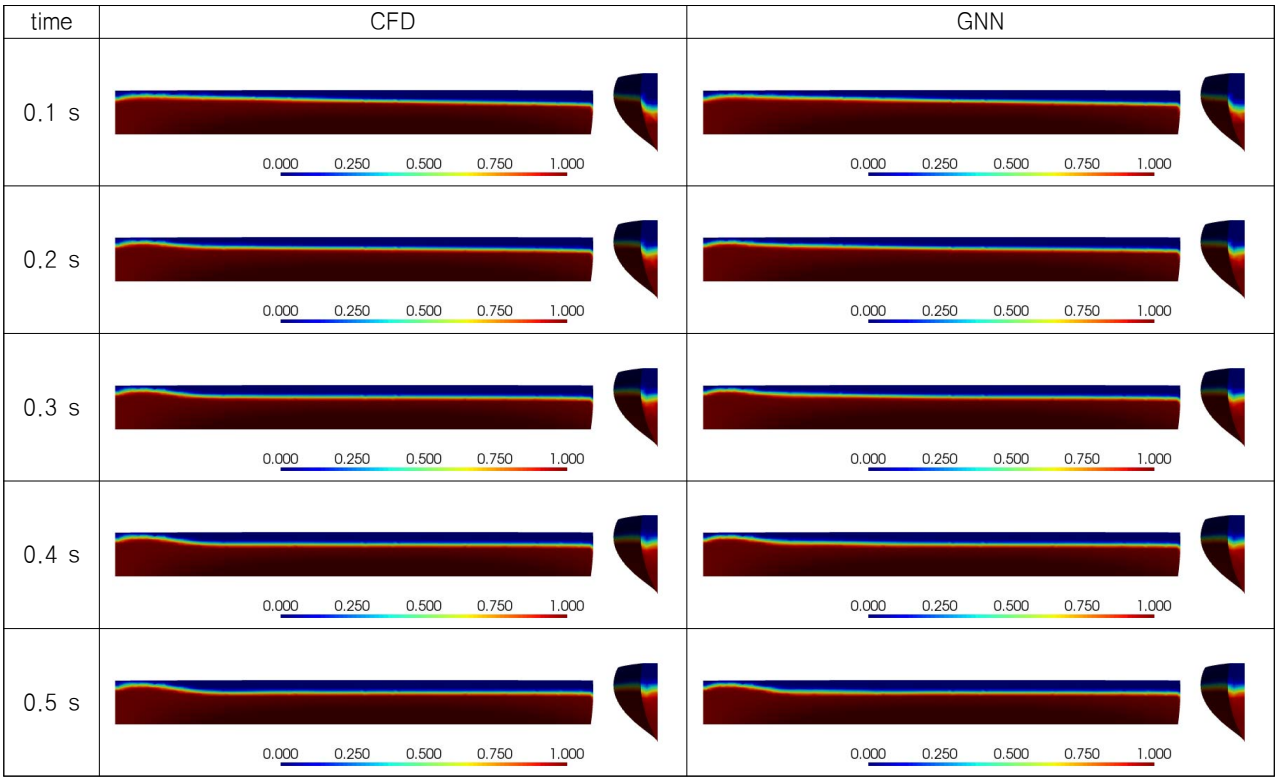
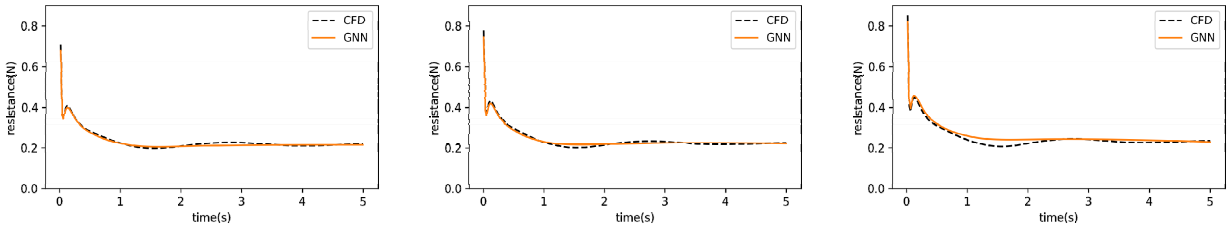


Fig. 8 Water volume fraction examples simulated with base breadth hull.



(a) −5 % breadth case (b) base breadth case (c) +5 % breadth case

Fig. 9 Comparison of resistance performance simulation using test set.

재 벡터를 출력하며, 2개의 은닉층과 SiLU(Sigmoid Linear Unit) 활성화 함수로 구현하였다 (Hendrycks and Gimpel, 2016). 개별적인 매개변수를 가지는 메시지 전달층은 8개를 사용하였다.

3.3 저항성능 추정 방법

시뮬레이션으로 계산한 선형의 저항 R은 압력 P와 전단응력 S의 합으로 계산했으며, P와 S는 각각 식 (7)과 식 (8)의 방식으로 계산하였다. 각 셀에 대해서 P_{cell}은 전압, A^x_{cell}은 x축 방향의 면적, S^x_{cell}은 x축 방향의 벽 전단응력, A_{cell}은 면적을 나타낸다.

$$P = \sum_{cell} P_{cell} \times A^x_{cell} \tag{7}$$

$$S = \sum_{cell} S^x_{cell} \times A_{cell} \tag{8}$$

여기서 x축은 시뮬레이션에서 유체가 흐르는 방향 축을 의미한다. GNN 모델을 이용한 실험은 시뮬레이션을 위한 초기 상태만 입력해서 다음 249번의 시간 단계 상태를 한 단계씩 연속적으로 예측하는 방식으로 구현하였다.

4. 결 과

Fig. 4~8은 시뮬레이션 초반부의 선형 표면에 걸리는 전압과 좌표축별 벽 전단응력, 물 부피분율 예시를 시간순으로 보여준다. 선체 표면에 GNN 모델이 예측한 물리량을 보면 선체의 옆 표면과 뒤 표면 모두에서 CFD 방식과 유사한 시뮬레이션이 가능한

것을 확인할 수 있다.

Fig. 9는 테스트용 데이터집합을 기존의 CFD 방식과 GNN 모델로 시뮬레이션한 저항성능을 시간 단계에 따라 비교한 결과를 보여준다. 시뮬레이션 과정에서 선형의 저항성능은 초기에 변화량이 크지만 시간이 지나면서 점차 안정화되어 정상상태로 수렴하는 형태를 보인다. GNN 모델은 시뮬레이션 초반 저항성능의 큰 변화를 잘 재현하지만, 이후에 작은 변화를 재현하지 못하는 것을 확인할 수 있다. 이는 GNN 모델이 물리량의 변화량을 학습하는 과정에서 상대적으로 변화량이 작은 부분을 학습하지 못한 것으로 보인다.

Table 3 Error of resistance prediction using test set (MAE*: mean absolute error over the entire time step).

case	R_p (CFD)	R_p (GNN)	error	MAE*
-5 %	0.21511	0.21396	0.535 %	2.493 %
0 %	0.22160	0.22367	0.933 %	2.721 %
+5 %	0.23067	0.23693	2.715 %	5.151 %
avg.	-	-	1.394 %	3.455 %

Table 3은 CFD 방식와 GNN 모델로 테스트용 데이터에 대해서 예측한 저항성능의 오류율을 보여준다. 저항성능 R_p는 시뮬레이션의 정상상태인 시간 3초에서 5초까지 저항의 평균값으로 계산하였으며, GNN 모델은 평균 1.4 %의 오차로 저항성능을 예측하였다. 시뮬레이션 전체 시간 단계에서의 오차는 평균 3.5 %로 너비를 +5 %로 조절한 경우에서 상대적으로 오차가 크게 나타나는 결과를 보였다.

5. 결 론

본 연구에서는 기존에 주로 사용하는 정규격자 방식이 아닌 메시 기반의 GNN 모델을 이용하여 선박 선형의 저항성능 시뮬레이션을 수행하였다. GNN 모델은 메시 형태의 기하학적 정보와 물리량을 그래프로 직접 표현할 수 있으며, 정규격자로는 표현하기 힘든 선형의 3차원 표면을 모델의 입력으로 사용할 수 있는 장점이 있다. CFD 시뮬레이션을 위해서 모델은 입력된 메시 상태의 한 단계 미래 상태를 예측하도록 학습하였으며, 학습된 모델을 자기회귀 방식으로 사용해서 연속적인 시뮬레이션 상태 시계열을 예측하였다.

비교적 단순한 선형에 대한 본 논문의 실험 결과로 다양한 종류의 복잡한 실제 선형에 대한 저항성능 시뮬레이션을 인공지능으로 수행하는 후속 연구의 가능성을 확인하였다. 하지만 다양한 조밀도의 격자를 가지는 일반적인 선형에 적용하기에는 모델이 학습한 데이터집합의 규모가 작으므로, 모델의 일반화를 위해서는 대규모 데이터집합을 대상으로 한 추가적인 연구가 필요하다. 또한 제안한 방법은 시뮬레이션을 수행하기 위해 최초 상태를 요구하는 제약사항이 있으며, 이 점을 해결하기 위한 새로운 방법이 필요하다고 생각된다.

후 기

이 연구는 산업통상자원부의 스마트특성화 기반구축사업 중 실물-가상연계 조선해양 기본설계 기술지원 사업 (P0021213) 과제 및 산업통상자원부와 한국산업기술진흥원의 지역혁신클러스터R&D 사업 (P0015330)의 지원을 받아 수행되었습니다.

References

Ba, J.L., Kiros, J.R. and Hinton, G.E., 2016. Layer Normalization. *arXiv preprint arXiv:1607.06450*.

He, K., Zhang, X., Ren, S. and Sun, Ji., 2016. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp.770-778.

Hendrycks, D. and Gimpel, K., 2016. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*.

Kim, B., Azevedo, V.C., Thuerey, N., Kim, T, Gross, M. and Solenthaler, B., 2019. Deep fluids: A generative network for parameterized fluid simulations. *Computer Graphics Forum*, 38(2), pp.59-70.

Kingma, D.P. and Ba, J., 2015. Adam: A method for stochastic optimization. *3rd International Conference for Learning Representations*, San Diego, May 7-9.

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A. and Battaglia, P.W., 2021. Learning mesh-based simulation with graph networks. *9th International Conference on Learning Representations*, Austria, May 3-7.

Thuerey, N., Weissenow, K., Prantl, L. and Hu, X., 2019. Deep learning methods for Reynolds-averaged navier-stokes simulations of airfoil flows. *American Institute of Aeronautics and Astronautics Journal*, 58(1), pp.25-36.


박 태 원


김 인 섭


이 훈


박 동 우