

공학석사학위논문

물리 기반 신경망을 이용한
탄성체의 거동 해석

Analysis of Continuous Elastic Structure
using Physics Informed Neural Networks

2022년 2월

서울대학교 대학원

생태조경·지역시스템공학부

지역시스템공학전공

정 영 준

물리 기반 신경망을 이용한 탄성체의 거동 해석

Analysis of Continuous Elastic Structure
using Physics Informed Neural Networks

지도교수 최 원

이 논문을 공학석사 학위논문으로 제출함
2022년 2월

서울대학교 대학원
생태조경·지역시스템공학부 지역시스템공학전공
정 영 준

정 영 준의 공학석사 학위논문을 인준함
2022년 1월

위 원 장 _____ (인)

부위원장 _____ (인)

위 원 _____ (인)

국문초록(16pt)

논문의 내용과 결론에 관하여 간략하고 구체적으로 기재

주요어 : (6단어 이내)

학 번 :

목 차

제 1 장 서론	1
제 1 절	1
제 2 절	2
 제 2 장 연구사	10
제 1 절	10
1.	10
2.	25
3.	20
제 2 절	30
1.	30
2.	35
3.	40
 제 3 장 재료 및 방법	100
3.1 탄성체의 거동 문제	100
.....	110
 참고문헌	120
Abstract	125

표 목 차

[표 1-1]	10
[표 2-1]	20

그 립 목 차

[그림 1-1]	15
[그림 2-1]	35
[그림 3-1]	55

제 1 장 서론

1.1 연구배경 및 필요성

물리적 현상을 해석하기 위해서는 먼저 수식으로 그 현상을 묘사할 필요가 있으며, 이것을 수학적 모델링이라고 한다. 수학적으로 모델링할 수 있는 현상들은 여러 가지가 있으며, 예를 들면 전염병의 전파, 인구 변화, 유체의 흐름, 구조체의 거동, 열 유동 등이 있다. 이런 현상들은 어떤 특성의 변화를 포함하고 있는 경우가 많은데, 예를 들어 특정 지역의 인구 변화를 모델링하면 시간에 따른 인구의 변화량이 수식에 포함될 것이며, 유체의 흐름을 모델링하면 유체의 속도가 수식에 포함될 것이다. 이러한 변화를 나타내기 위해 미분항이 수식에 포함될 수 밖에 없고, 따라서 수학적 모델링은 미분방정식으로 주로 이루어진다. (참고문헌)

수학적 모델링을 통해 얻은 미분방정식의 해를 구하면 비로소 수학적 으로 그 현상을 해석했다고 말할 수 있으나, 실제로는 해석적인 해를 구할 수 없는 미분방정식이 많이 존재한다. 이러한 미분방정식들의 해를 구하기 위해서는 수치해석 (Numerical analysis) 기법을 사용하여 엄밀 해에 가까운 근사해를 구해야 한다 (Chapra and Canale, 2011). 특히, 미분방정식의 독립변수가 2개 이상이고, 2개 이상의 독립변수에 대한 미분항이 포함되는 편미분방정식은 엄밀해를 구하기가 매우 어렵기에 수치해석 기법을 사용하여 근사해를 구하는 경우가 대부분이다. 편미분방정식의 근사해를 구하는 방법으로는 FDM (Finite differential method), FEM (Finite element method), BEM (Boundary element method) 등이 있다. (참고문헌)

한편, 심층신경망 (DNN; Deep neural networks)을 이용하는 기계학습 알고리즘인 딥러닝은, 이미지 인식, 음성 인식 등에 성공적으로 사용되었을 뿐만 아니라, 유체역학, 재료과학, 유전공학 등 다양한 공학 및 과학 분야의 문제를 해결하는 데에도 광범위하게 적용되고 있다 (LeCun et al., 2015; Libbrecht and Noble, 2015; Rafiei and Adeli, 2017; Raissi

et al., 2019). 딥러닝은 최근 물리적 시스템의 수학적 모델링과 시뮬레이션에도 적용되고 있는데, 이는 복잡한 비선형 함수를 학습할 수 있는 심층신경망의 특성 때문이다 (Rao et al., 2021).

심층신경망을 사용해서 물리적 시스템의 지배방정식을 해석하는 방법으로 물리 기반 신경망 (Physics-informed neural networks; PINN)가 주목받고 있다. PINN은 심층신경망의 일종으로, 미분방정식의 경계조건, 초기조건, 그리고 지배방정식을 학습하여 미분방정식의 근사해를 구하는 방법이다. 이 방법은 전통적인 수치해석 방법과 비교했을 때, 격자를 생성하지 않아도 근사해를 구할 수 있으며 해의 미분값들을 다루기 쉽게 제공해준다는 점에서 강점이 존재한다 (Raissi et al., 2019). 그러나 PINN 역시 모든 미분방정식을 해석할 수 있는 것은 아니며, 특히 낮은 주파수를 가진 해보다 높은 주파수를 가진 해를 구할 때 수렴이 훨씬 오래 걸린다는 문제가 있으며, 이를 스펙트럼 편향 (Spectral bias)라고 한다 (Xu et al., 2019, Rahaman et al., 2019, Basri et al., 2019, Cao et al., 2019).

일반적인 PINN의 연산 속도나 수렴성을 개선하기 위한 여러 시도가 있어왔으며, 영역 분할 (Domain decomposition)은 이러한 시도 중 하나이다. 영역 분할은 전체 연산 영역 (Computational domain)을 소영역 (Subdomain)으로 분할하여 각 소영역에서 신경망을 각각 학습시켜 국부적인 해를 구하되, 전체 영역에서의 해에 대한 구속조건을 학습에 반영하여 전체적인 해를 구하는 방법이다. 영역 분할은 PINN으로 근사해를 구할 때의 복잡도를 감소시키고, 연산 시간을 단축시킬 수 있는 방법으로 보고된 바 있다 (Jagtap et al., 2020; Shukla et al., 2021).

한편, PINN은 비교적 최신 기술이고 최근에 연구가 왕성하게 이루어지고 있는 분야이다. 최근 들어 유체역학, 균열 전파 등 특정 주제의 문제들의 해석에 대한 연구가 활발히 진행되고 있으나 아직 구조해석 분야에서 그 수가 많지는 않은 상황이다 (Sun et al., 2020; Goswami et al., 2020).

1.2 연구목적

본 연구에서는 다양한 구조해석 문제를 PINN을 사용해서 해석하고, 일반적인 기존 PINN이 해석하지 못하는 문제에 대하여 영역 분할을 적용하여 해석한다. 또한 기존 PINN으로 해석할 수 있는 문제에 대하여 영역 분할을 적용하였을 때의 학습 성능에 대해 비교하고 분석한다. 본 연구의 흐름은 Fig. 1.1과 같으며 크게 세 단계로 구분된다. 먼저 일반적인 PINN으로 해석할 수 있는 구조해석 문제를 해석하고, 해석 결과와 학습 결과를 제시한다. 다음으로 일반적인 PINN으로 해석할 수 없는 구조해석 문제에 대하여 영역 분할을 적용한 새로운 형태의 PINN을 제시하고, 이를 적용하여 해석 결과와 학습 결과를 제시한다. 마지막으로 일반적인 PINN으로 해석할 수 있는 구조해석 문제에 대하여 영역 분할을 적용한 새로운 형태의 PINN을 적용했을 때의 해석 결과와 학습 결과에 대해 비교하여 제시한다. (Flow chart 추가 필요)

본문은 38자 × 24행 정도로 인쇄한다.

분문 분량은 논문 규격(19cm × 26cm)을 기준으로 자유롭게 한다.

본문내용 : 글자크기 11pt 정도, 줄간격 170이상, 장평 100, 자간 0

각주 : 글자 크기 9~10pt - 각주 예시¹⁾

서체 : 명조체나 신명조체 류

글씨색 : 흑색(수록 자료는 칼라 가능)

용지 여백 : 위쪽 20, 아래쪽 15, 머리말 15, 꼬리말 15, 왼쪽 30, 오른쪽 30

페이지 표시는 각 면 하단 중앙에 위치하게 한다.

1)각주 내용

제 2 장 연구사

2.1 PINN

PINN의 개념을 처음 제안한 것은 Lagaris et al. (1998)이며, 이 연구에서는 인공신경망에 미분방정식의 초기조건 및 경계조건을 학습시켜 근사해를 구한 후 갤러킨 방법을 적용한 FEM을 통해 구한 근사해와 비교하였다. 한편 다른 인공신경망 관련 연구가 그렇듯, 최근 들어 개선된 하드웨어가 출시되고 사용하기 쉬운 딥러닝 프레임워크가 개발되는 등 여건이 개선됨에 따라 개념으로만 제시되는 수준에 그쳤던 PINN 역시 Raissi et al. (2019)에 의해 최신 딥러닝 기술을 적용하여 구현되었다.

Raissi et al. (2019)은 해당 연구에서 PINN을 사용하여 순방향 문제 (Forward problem)와 역방향 문제 (Inverse problem)를 해결하였다. 순방향 문제는 어떤 현상에 대하여 알려진 지배방정식과 초기조건 및 경계조건이 주어져 있을 때 PINN을 사용하여 근사해를 구하는 문제를 의미하고, 역방향 문제는 반대로 어떤 현상의 지배방정식과 초기조건, 경계조건이 완전하게 알려져 있지 않은 상황에서 그 현상의 실측 데이터를 바탕으로 지배방정식, 초기조건, 경계조건을 추론하는 문제이다. Raissi et al. (2019)는 PINN을 사용하여 슈뢰딩거 방정식 (Schrodinger equation)과 알렌-칸 방정식 (Allen-Cahn equation)에 대하여 순방향 문제를 해결한 바 있으며, 나비에-스토크스 방정식 (Navier-Stokes equation)과 코르테버흐-더프리스 방정식 (Korteweg-de Vries equation)에 대하여 역방향 문제를 해결하여 알려지지 않은 지배방정식의 계수를 추정해낸 바 있다.

Raissi et al. (2019) 이후의 연구들은 PINN의 형태를 변경하고 새로운 형태를 제안하여 해석 성능을 개선하거나, 유체역학, 균열 전파 등 특정 주제의 문제의 해석에 대한 연구가 주를 이룬다. Mao et al. (2020)은 1차원 및 2차원 정상 상태에서의 오일러 방정식 (Euler equation)의 근사해를 PINN을 사용해 구함으로써 압축 비점성 유체의 흐름을 모의하였

다. Chen et al. (2020)은 PINN을 사용해 광자 메타 물질 (Photonic metamaterial)과 나노광학 기술에서의 산란 문제에 관련된 지배방정식의 알려지지 않은 계수를 추정하였다. Samaniego et al. (2020)은 PINN을 사용하되 함수 최적화에 에너지법을 적용하여 선형 탄성체의 거동을 모의한 바 있다. Sun et al. (2020)은 PINN을 적용하되 경계조건 강제 (Boundary condition enforcement)를 사용하여 나비에-스토크스 방정식을 해석하였고 이를 통하여 원형 관에서의 흐름, 혈액의 흐름 등 다양한 유체 흐름 문제를 해결한 바 있다. Goswami et al. (2020)은 위상-필드 모델 (Phase-field model) 문제를 PINN을 사용해 해석하여 균열의 전파를 모의한 바 있으며, PINN의 전이학습 (Transfer learning) 적용 가능성을 제시하였다. Haghighat et al. (2021)은 고체 역학 (Solid mechanics) 문제에 PINN을 적용하여 2차원 구조체의 응력 및 처짐 분포를 모의한 바 있으며, Niaki et al. (2021)은 복합 재료 (Composite material)의 열화학적 복원 (Thermochemical curing) 문제를 PINN을 적용해 해석한 바 있다.

2.2 영역 분할을 적용한 PINN

PINN의 수렴 속도를 높이고 최적화 과정의 복잡도를 낮추기 위하여 영역 분할을 적용한 PINN이 연구되고 있다 (Jagtap et al. 2020; Heinlein et al., 2021). PINN에 영역 분할을 적용하여 소영역들로 전체 영역의 근사해를 구하는 과정을 학습시킬 때 유의해야 할 점은, 대부분의 해가 연속적이기에 각 인접한 소영역들의 경계인 인터페이스에서 근사해의 값의 연속성을 보장해야 하며, 동시에 전체 영역에서의 지배방정식, 경계조건 및 초기조건 역시 만족해야 한다는 점이다.

Jagtap et al. (2020)은 cPINN (Conservative physics-informed neural network)을 제시하였으며, 영역을 분할하여 각 소영역에 개별적인 PINN을 배정하고, 소영역 간의 인터페이스에서 좌우 소영역의 플럭스 연속성을 보장하는 오차항을 손실함수로 포함시켜 각 소영역의 PINN을 학습시켜 전체 영역의 근사해를 모의하도록 하였다. 이후

Jagtap and Karniadakis (2020)은 cPINN에서 더 발전된 형태인 XPINN (Extended physics-informed neural networks)을 제시하였다. XPINN은 cPINN이 아무 편미분방정식에나 적용될 수 없다는 한계를 극복하여 모든 종류의 편미분방정식에 적용될 수 있으며, 영역 분할에 제약조건이 있는 cPINN과 달리 시간과 공간 모두에서 임의의 형태로 분할할 수 있다는 이점이 존재한다. Shukla et al. (2021)은 cPINN과 XPINN을 다수의 CPU와 GPU로 분산시켜 동시에 학습시키는 접근법을 보여주었으며, 단일 CPU 혹은 GPU에서 학습시킨 cPINN 및 XPINN보다 연산 속도가 증가함을 보고한 바 있다.

이외에도 Kharazmi et al. (2021)은 변분법 (Variational formulation)과 페트로프-갈러킨 방법을 적용하여 연산 영역을 분할한 hp-VPINN (hp-Variational physics-informed neural network)을 제안하였고, 다양한 문제에서 그 적용 결과를 제시하였다. Moseley et al. (2021)은 연산 영역이 큰 문제에서 발생하는 스펙트럼 편향을 해소하기 위한 FBPINN (Finite basis physics-informed neural network)을 제안하였다. 이는 기존 방법과 다르게 영역을 겹치는 영역이 있게 분할하며, 이때 발생하는 인터페이스 영역에서 윈도우 함수를 적용하여 cPINN이나 XPINN 방법을 적용했을 때 소영역의 근사해 간의 오차로 인해 발생하는 불연속성 문제를 해결하고 전체 영역에서 연속성을 보장하였다.

2.3 구조해석에 사용된 PINN

Raissi et al. (2019) 이후, 각 분야에서 PINN을 활용하여 모델을 해석하거나 미지의 지배방정식을 추론하는 연구들이 활발히 수행되었으며, 그 중 구조해석에 관련된 연구도 더러 있다.

Samaniego et al. (2020)은 에너지법을 적용한 PINN을 구현하여 선형 탄성 재료와 고탄성 재료 (Hyperelastic material)의 처짐, 위상-필드 모델을 이용한 균열 전파 양상, 그리고 판의 휨 (Plate bending)과 같은 다양한 구조해석 문제를 해석하고 결과를 제시한 바 있다. Haghighat et al. (2021)은 Raissi et al. (2019)이 제시한 PINN을 사용하여 2차원 평면

-응력 문제 (Plane-stress problem)을 해석하였고 재료에서의 처짐과 응력 분포를 모의한 바 있다. 이때 각 처짐과 응력 요소를 별개의 모델에서 학습시켜 총 5개의 PINN을 학습시킨 것이 특징적이다. Rao et al. (2021)은 PINN의 구조를 변형하여 경계조건을 강제하였고 2차원 평면-응력 문제를 해석한 바 있으며, Haghighat et al. (2021)과 마찬가지로 여러 개의 PINN을 사용하여 처짐과 응력의 개별 요소를 각각 모의하였다. 이외에도 Nabian et al. (2021)은 PINN의 학습 데이터에 중요도 샘플링 (Importance sampling)을 적용하여 2차원 평면 응력 문제의 수렴 속도를 개선한 바 있다.

제 3 장 재료 및 방법

3.1 탄성체의 거동 문제

3.1.1 1차원 보 (1-D Beam)

대부분의 보의 처짐은 매우 작으며 곡률도 아주 작기 때문에, 보의 회전각 θ 과 보의 기울기 $\frac{dv}{dx}$ 가 같다고 둘 수 있고, 보의 곡률은 $\frac{d\theta}{dx}$ 와 같으며 이는 $\frac{d^2v}{dx^2}$ 와 같다. 만약 보의 재료가 선형 탄성적이라고 가정하면 후크 법칙 (Hooke's law)를 따르게 되고, 이때 보의 곡률은 $\frac{M}{EI}$ 와 같게 된다 (Goodno and Gere, 2017). 이를 통해 식 (1)과 같이 굽힘모멘트와 보의 처짐에 관한 미분방정식을 얻을 수 있으며, 식 (1)의 전단력, 굽힘모멘트, 분포하중에 대한 관계로부터 처짐과 분포하중에 관한 미분방정식인 식 (2)를 얻을 수 있다.

	$\frac{d^2u}{dx^2} = \frac{M}{EI}$ $\frac{dV}{dx} = w$ $\frac{dM}{dx} = V$	(1)
--	--	-----

	$EI \frac{\partial^4 u}{\partial x^4} - w = 0$	(2)
--	--	-----

where, u is a displacement of the beam, M is a bending moment of the beam, E is Young's modulus, I is a moment of inertia, V is a shear force, w is a distributed load.

따라서 1차원 보에 임의의 하중이 작용할 때 지배방정식은 식 (2)와

같다. 이를 풀기 위해서는 경계조건이 필요하며, 경계조건은 지지점의 종류에 따라 달라지게 된다. 예를 들어 단순보의 양 끝 지지점 같은 경우에는 처짐과 굽힘모멘트가 0이므로 식 (3)과 같은 경계조건이 주어지고, 연속보에서 양 끝 지지점이 아닌 가운데에 있는 지지점의 경우에는 단순히 처짐만 0이므로 식 (4)와 같은 경계조건이 주어진다. 캔틸레버보의 고정 지지점의 경우에는 처짐과 기울기가 0이므로 식 (5)와 같이 경계조건이 설정되고, 자유단의 경우에는 굽힘모멘트와 전단력이 0이므로 식 (6)과 같은 경계조건이 설정된다.

	$u(a) = u_{xx}(a) = 0$	(3)
--	------------------------	-----

	$u(a) = 0$	(4)
--	------------	-----

	$u(a) = u_x(a) = 0$	(5)
--	---------------------	-----

	$u_{xx}(a) = u_{xxx}(a) = 0$	(6)
--	------------------------------	-----

where, a is a boundary point, $u(a)$ is a displacement of the beam at $x=a$, and u_x, u_{xx}, u_{xxx} are the first, second, and third derivatives of u .

3.1.2 2차원 평면 응력 (2-D Plane Stress)

어떤 재료의 응력 요소들이 식 (7)과 같을 때 평면 응력 (Plane stress) 상태라고 말할 수 있다. 이때 평형 방정식 (Equilibrium equation)은 식 (8)과 같고, 변형률-응력 관계, 변형률-변위 관계는 각각 식 (9), 식 (10)과 같다 (Chou and Pagano, 1967).

	$\begin{aligned}\sigma_x &= \sigma_x(x, y) \\ \sigma_y &= \sigma_y(x, y) \\ \tau_{xy} &= \tau_{xy}(x, y) \\ \tau_{xz} &= \tau_{yz} = \sigma_z = 0\end{aligned}$	(7)
--	---	-----

	$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + f_x = 0$ $\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + f_y = 0$	(8)
--	---	-----

	$\epsilon_x = \frac{1}{E}(\sigma_x - \nu \sigma_y)$ $\epsilon_y = \frac{1}{E}(\sigma_y - \nu \sigma_x)$ $\gamma_{xy} = \frac{1}{G} \tau_{xy}$	(9)
--	---	-----

	$\epsilon_x = \frac{\partial u}{\partial x}$ $\epsilon_y = \frac{\partial v}{\partial y}$ $\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$	(10)
--	---	------

where, σ_x, σ_y are normal stresses in x and y direction and τ_{xy} a shearing stress, $\epsilon_x, \epsilon_y, \gamma_{xy}$ are normal strains in x and y direction and a shearing strain, f_x, f_y are body forces in x and y direction. G 추가, u, v are displacements in x and y direction.

식 (9)와 식 (10)에서 변형률 항을 소거하고 식 (8)에 대입하면 식 (11), (12), (13)과 같은 x, y 방향 처짐과 힘에 대한 미분방정식을 얻을 수 있다.

	$G\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + (\lambda + G)\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + f_x = 0$ $G\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + (\lambda + G)\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + f_y = 0$	(11)
--	---	------

	$\lambda = \frac{\nu E}{(1 + \nu)(1 - \nu)}$	(12)
--	--	------

	$G = \frac{E}{2(1+\nu)}$	(13)
--	--------------------------	------

where, ν is the Poisson's ratio and E is the Young's modulus of the structure.

3.1.3 2차원 판 (2-D Plate)

3차원 물체의 z 방향 높이가 매우 작아 무시할 수 있을 때 이를 2차원 판으로 근사할 수 있다. 2차원 판의 굽힘모멘트와 비틀림모멘트의 관계식은 식 (14)와 같으며, 모멘트와 하중의 관계는 식 (15)와 같다. 식 (14)와 식 (15)를 합치면 식 (16)과 같이 2차원 판에서 하중과 처짐에 관한 미분방정식을 얻을 수 있다 (Timoshenko and Woinowsky-Kreiger, 1959; Ugural, 2009).

	$\frac{\partial^2 M_x}{\partial x^2} + 2\frac{\partial M_{xy}}{\partial x \partial y} + \frac{\partial^2 M_y}{\partial y^2}$	(14)
--	--	------

	$\begin{aligned} M_x &= -D \left(\frac{\partial^2 w}{\partial x^2} + \nu \frac{\partial^2 w}{\partial y^2} \right) \\ M_y &= -D \left(\frac{\partial^2 w}{\partial y^2} + \nu \frac{\partial^2 w}{\partial x^2} \right) \\ M_{xy} &= -M_{yx} = D(1-\nu) \frac{\partial^2 w}{\partial x \partial y} \end{aligned}$	(15)
--	---	------

	$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} - \frac{q}{D} = 0$	(16)
--	---	------

	$D = \frac{Eh^3}{12(1-\nu^2)}$	(17)
--	--------------------------------	------

M_x, M_y, M_{xy} are bending moments in x, y direction and a shearing moment, D is the flexural rigidity of the plate, defined in Eq. (17). ν is Poisson's ratio, h is a depth of the plate, w is

a displacement in z direction.

3.2 인공신경망을 이용한 미분방정식의 해석

3.2.1 인공신경망

인공신경망 (Artificial neural networks)은 생물의 신경망에서 영감을 받아 만들어진 학습 알고리즘으로, 패턴 인식, 예측, 최적화 등 다양한 분야에서 사용되고 있는 방법이다 (Jain et al., 1996). 인공신경망은 입력층 (Input layer), 은닉층 (Hidden layer), 그리고 출력층 (Output layer)으로 구성되고, 각 층은 개개의 뉴런 (Neuron)들로 구성되어 있다. 입력층에서 들어온 데이터를 바탕으로 신경망의 출력층의 출력 데이터가 계산되면, 이를 정답 데이터와 비교하여 오차를 계산 후 오차를 줄이는 방향으로 각 뉴런의 가중치와 편향을 업데이트한다. 이때 출력 데이터 계산 시의 방향과 반대로 업데이트가 이루어지기 때문에 이를 역전파 (Back propagation)이라고 일컫는다 (LeCun et al., 2015).

각 뉴런에는 고유한 가중치 (Weight)와 편향 (Bias)가 있어, 이전 층의 뉴런으로부터 전달된 값에 가중치를 곱한 후 편향을 더해서 비선형 활성화 함수 (Nonlinear activation function)에 대입하여 다음 층으로 전달하는 역할을 한다 (Nielsen, 2015). 특정 뉴런의 값은 식 (18)과 같이 나타낼 수 있다. l 번째 층의 j 번째 뉴런은, 직전 층인 $l-1$ 번째 층의 모든 뉴런과 가중치를 곱하고 편향을 더하여 활성화함수 σ 에 대입한 것과 같다. 이때 w_{jk}^l 은 $l-1$ 번째 층의 k 번째 뉴런과 l 번째 층의 j 번째 뉴런 사이의 가중치이며, b_j^l 은 l 번째 층의 j 번째 뉴런의 편향을 나타낸다.

	$a_j^l = \sigma(\sum_k w_{jk}^l a_k^{l-1} + b_j^l)$	(18)
--	---	------

N 번째 층의 가중치 벡터를 w^N , 편향 벡터를 b^N , 신경망의 목적함수를 C 라고 하면 식 (19)와 같이 나타낼 수 있다. 신경망의 편향과 가중치

는 경사하강법을 사용하여 최적화하게 되는데, 경사하강법은 어떤 함수의 기울기를 구하여 극값에 도달할 때까지 계속해서 기울기의 반대 방향으로 이동시키는 방법을 말한다. 경사하강법을 사용해 각 층의 편향과 가중치의 값을 목적함수의 크기를 작게 하는 방향으로 업데이트하는 것이 식 (20), (21)에 나타나 있으며, 목적함수의 N 번째 층 편향에 대한 기울기는 식 (22), 가중치에 대한 기울기는 식 (23)과 같이 $N+1$ 번째 층의 값을 통해 계산할 수 있다. 이처럼 신경망의 출력 값을 구할 때와 반대 방향으로 출력층에서 입력층으로 차례차례 업데이트하기 때문에 이 과정을 역전파라고 말한다.

	$s^N = w^N a^{N-1} + b^N, \quad a^N = \sigma(s^N), \quad \frac{\partial C}{\partial s^N} = \delta^N$	(19)
--	--	------

	$b^N \leftarrow b^N - \eta \frac{\partial C}{\partial b^N}$	(20)
--	---	------

	$w^N \leftarrow w^N - \eta \frac{\partial C}{\partial w^N}$	(21)
--	---	------

where, η is the learning rate.

	$\begin{aligned} \frac{\partial C}{\partial b^N} &= \frac{\partial C}{\partial s^N} \frac{\partial s^N}{\partial b^N} = \frac{\partial C}{\partial a^N} \frac{\partial a^N}{\partial s^N} \frac{\partial s^N}{\partial b^N} \\ &= \frac{\partial C}{\partial s^{N+1}} \frac{\partial s^{N+1}}{\partial a^N} \frac{\partial a^N}{\partial s^N} \frac{\partial s^N}{\partial b^N} = \delta^{N+1} w^{N+1} \sigma'(s^N) \end{aligned}$	(22)
--	--	------

	$\begin{aligned} \frac{\partial C}{\partial w^N} &= \frac{\partial C}{\partial s^N} \frac{\partial s^N}{\partial w^N} = \frac{\partial C}{\partial a^N} \frac{\partial a^N}{\partial s^N} \frac{\partial s^N}{\partial w^N} \\ &= \frac{\partial C}{\partial s^{N+1}} \frac{\partial s^{N+1}}{\partial a^N} \frac{\partial a^N}{\partial s^N} \frac{\partial s^N}{\partial w^N} = \delta^{N+1} w^{N+1} \sigma'(s^N) a^{N-1} \end{aligned}$	(23)
--	--	------

3.2.2 PINN

PINN은 심층 신경망의 일종으로서, 다음과 같은 미분방정식 (식 (1),

(2))를 만족하는 해 u 의 근사해 \hat{u} 를 구하기 위하여 사용된다.

	$N[u(x)] = 0, x \in D$	(24)
--	------------------------	------

	$B[u(x)] = 0, x \in \partial D$	(25)
--	---------------------------------	------

where, N is a differential operator, B is a boundary operator, D is a computational domain.

PINN이 기존의 심층 신경망과 가장 다른 점은, 학습에 있어서 식 (1), (2)와 같이 물리현상을 나타내는 미분방정식을 적용할 수 있다는 것이다. 근사해 \hat{u} 를 신경망을 통하여 나타내면 식 (3)과 같고, 신경망의 학습을 위한 목적함수인 손실 (Loss)는 식 (4)와 같이 나타낼 수 있다. PINN의 손실은 경계조건에 대한 손실과 지배방정식에 대한 손실의 합으로 나타낼 수 있다.

	$\hat{u} = NN(x; \theta)$	(26)
--	---------------------------	------

	$L(x; \theta) = L_B + \lambda L_F = B[\hat{u}(x; \theta)] ^2 + \lambda N[\hat{u}(x; \theta)] ^2$
--	--

where, NN is a neural network, θ is parameters of neural network (e.g. weights, biases), L_B is a loss regard to boundary conditions, L_F is a loss regard to governing differential equations, λ is a penalty parameter.

결국 PINN의 학습을 통하여 주어진 미분방정식의 근사해를 구하는 과정은 식 (5)와 같은 비구속 최적화 문제로 바꿀 수 있다. 손실의 값을 최소화하는 신경망의 최적 모수 θ^* 를 신경망의 역전파를 통하여 찾아 나가게 된다. 예를 들어 2차원 정상 (Stationary) 문제를 해결하기 위한 PINN의 구성도를 Fig. 1과 같이 나타낼 수 있다.

	$\theta^* = \operatorname{argmin}_{\theta} (\ B[\hat{u}(x;\theta)]\ ^2_{\partial D} + \lambda \ N[\hat{u}(x;\theta)]\ ^2_D)$	(28)
--	--	------

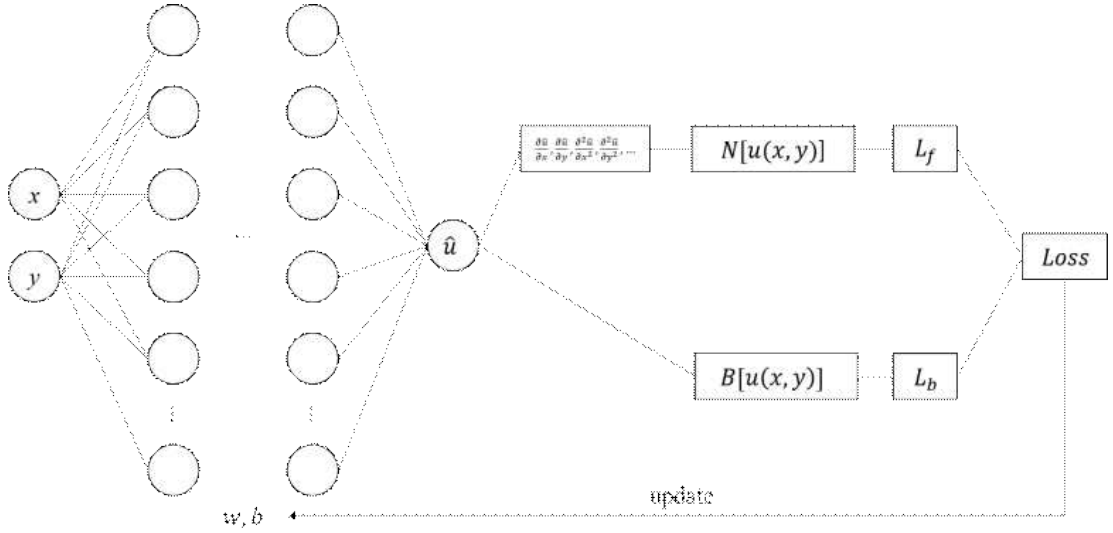


Fig. 1 Schematic of PINN solving 2-dimensional stationary problems

PINN을 학습시키기 위한 데이터의 추출 방식을 Fig. 에 나타내었다. 먼저 경계조건의 학습을 위하여 ∂D 에서 임의의 개수의 데이터를 추출한다. 추출한 데이터를 신경망에 입력하고 출력되는 값과 경계조건에 주어진 경계값 간의 오차를 계산한다. 다음으로 지배방정식의 학습을 위하여 D 에서 임의의 개수의 데이터를 추출하여, 그 데이터를 신경망에 입력하고 신경망의 미분값을 계산한 뒤 지배방정식에 대입하여 오차를 계산한다.

예를 들어 식 ()과 같은 미분방정식을 PINN을 통해 해석한다고 가정해보자. 이 경우 L_B , L_F 는 각각 식 (), 식 ()과 같이 정의된다.

	$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0,$ $u(0,y) = 0, \quad u(1,y) = 10, \quad u(x,0) = 0, \quad u(x,1) = 0$	(29)
--	--	------

	$L_B = (\hat{u}(0,y) - u(0,y))^2 + (\hat{u}(1,y) - u(1,y))^2 + (\hat{u}_y(x,0) - u_y(x,0))^2 + (\hat{u}_y(x,1) - u_y(x,1))^2$	(30)
--	---	------

	$L_F = (\hat{u}_{xx}(x,y) + \hat{u}_{yy}(x,y) - 0)^2$	(31)
--	---	------

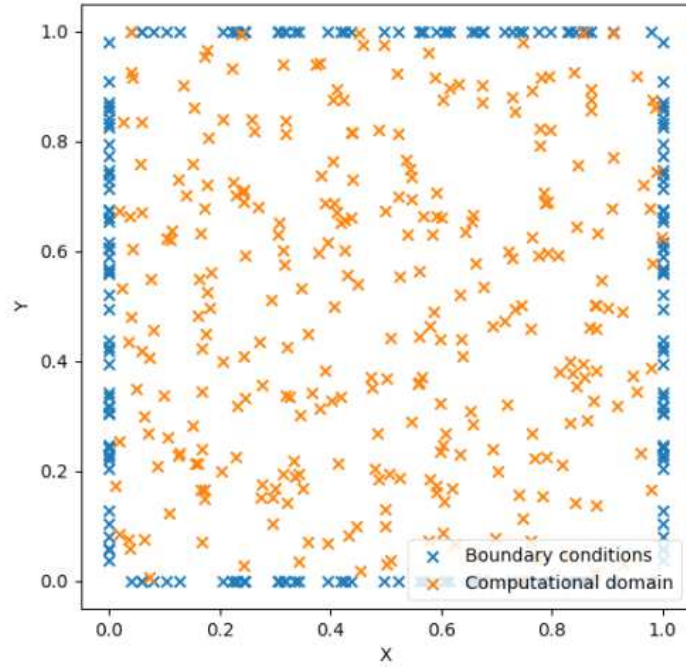


Fig. 2 Sampling of the training data of PINN from boundary conditions and computational domain

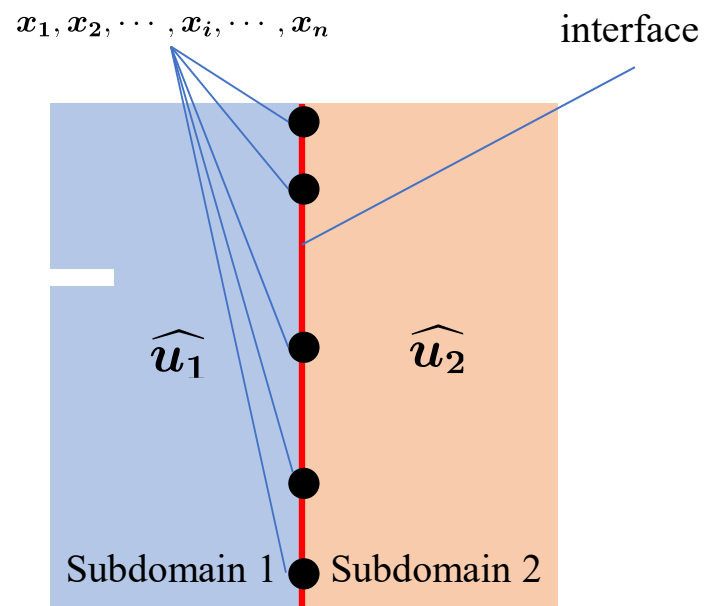
3.3 Piecewise PINN

PINN는 FDM, FEM 등 고전적인 수치해석 방법과 달리 격자망을 구성할 필요가 없어 연산 영역의 형상이 복잡하거나 큰 경우에 이점을 발휘할 수 있으며, 격자망에 해당하는 해만 구하는 FDM, FEM 등과 달리 전체 연산 영역에 대하여 근사해를 제공한다. 그러나 앞에서 밝혔듯 일

반적인 PINN 만으로는 해석하지 못하는 문제도 더러 있으며, 본 연구에서는 그런 문제를 해결하기 위하여 piecewise PINN이라는 새로운 형태의 PINN을 제안한다.

Piecewise PINN의 학습 방식은 기존 PINN과 매우 유사하게 분해된 소영역 (Subdomain)들의 경계조건과 지배방정식에 대한 손실을 구하여 학습하며, 여기에 소영역들 간의 연속성을 담보하기 위한 손실 (Interface Loss; L_I)이 추가된다. 어떤 소영역 S 과 인접한 소영역과의 경계를 생각해 보면, 그 경계에서의 근사해는 소영역 S 의 신경망뿐만 아니라 인접한 소영역의 신경망으로도 계산할 수 있다. 각 소영역의 신경망으로 구한 경계에서의 근사해와 1차, 2차, 3차, ..., n 차 미분계수 간의 오차를 손실로 두고 이를 0으로 만드는 방향으로 최적화를 진행한다면 각 소영역 간의 연속성과 미분 가능성을 보장할 수 있다.

Piecewise PINN을 학습시키기 위한 데이터의 추출 방식을 Fig. 에 나타내었다. 경계조건과 지배방정식의 충족을 위한 학습 방식은 PINN의 그것과 동일하다. 추가된 것은, 소영역 간의 연속성과 미분가능성을 성립시키기 위한 L_I 이다. 이는 인접한 소영역들의 경계인 인터페이스에서 임의의 개수의 데이터를 추출하고, 그 데이터를 각 소영역의 신경망에 입력한 뒤 출력되는 값과 1차, 2차, 3차, ..., n 차 미분계수의 차이로 정의된다. 이때 n 의 값은 해석하고자 하는 문제의 종류나 차원에 따라 달라질 수 있다.



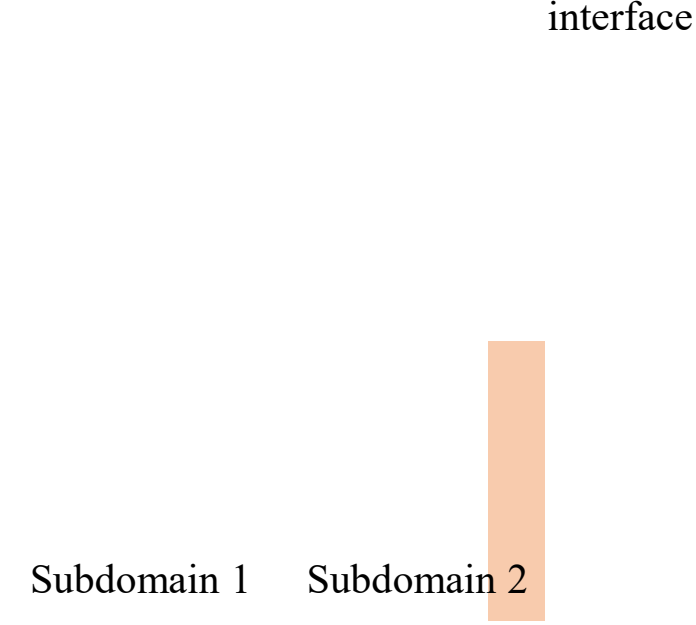


Fig. 5 Sampling of the training data of piecewise PINN from an interface between two subdomains

	$F[\mathbf{u}(\mathbf{x})] = (\mathbf{u}, \frac{\partial \mathbf{u}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{u}}{\partial \mathbf{x}^2}, \frac{\partial^3 \mathbf{u}}{\partial \mathbf{x}^3}, \dots)$	(32)
--	--	------

	$L_I = \ F[\hat{\mathbf{u}}(\mathbf{x}^+; \boldsymbol{\theta})] - F[\hat{\mathbf{u}}(\mathbf{x}^-; \boldsymbol{\theta})]\ _I^2$	(33)
--	---	------

where, I is a set of interfaces, which means boundary of two adjacent subdomains, \mathbf{x}^+ and \mathbf{x}^- are the values of the argument on the right and left hand of an interface between adjacent subdomains.

함수 자체와 그 미분값들이 주어진 영역 전체에서 연속이어야 하는 경우에는 Eq. 7과 같이 손실을 설정하여야 하나, 만약 어떤 지점을 경계로 함수값이나 그 미분값이 특정 값만큼 차이나 불연속이 되는 경우가 있을 수 있다. 그 예시로 1차원 보 문제에서 어떤 점 x 에 집중하중 P 가 주어지는 경우를 생각해 보면, 집중하중이 주어지는 점 x 좌우에서 전단력이 P 만큼 차이나게 되고 전단력은 x 에서 불연속이 된다. 이런 경우에

는 L_I 를 계산할 때 경계 좌우로 함수값 또는 미분값의 차이와 P 의 차이를 계산하여 반영해주면 된다. 따라서 Eq. 7을 더 일반적인 경우에 대해 쓰면 Eq. 8과 같다.

	$L_I = \ F[\hat{\mathbf{u}}(\mathbf{x}^+; \boldsymbol{\theta})] - F[\hat{\mathbf{u}}(\mathbf{x}^-; \boldsymbol{\theta})] - \mathbf{P}\ _I^2$	(34)
--	--	------

where, P is a vector regard to the difference between \mathbf{x}^+ and \mathbf{x}^- . If the n th derivative of the neural network is continuous at the interface, the $(n+1)$ th element of P is zero, and if it is discontinuous, it has nonzero value.

제 4 장 결과 및 고찰

4.1 1차원 보 문제의 해석

4.1.1 문제의 설정

가장 먼저 1차원 보의 처짐을 해석하기 위하여 일반 PINN을 적용하였다. 신경망의 구조는 뉴런 1개로 구성된 입력층 (Input Layer)과 마찬가지로 뉴런 1개로 구성된 출력층 (Output Layer), 그리고 뉴런 40개로 구성된 은닉층 (Hidden Layer) 3개로 구성하였다. 옵티마이저 (Optimizer)로는 각 파라미터마다 다른 크기의 업데이트를 적용하는 Adam 옵티마이저를 사용하였으며, 층 간의 활성화 함수로는 쌍곡탄젠트 (Hyberbolic Tangent) 함수를 사용하였다. 학습률 (Learning rate)로는 0.0002를 적용하였으며 총 2000번의 에폭 (Epochs) 동안 학습을 진행하였다. 학습 데이터는 경계조건에 대하여 100개, 지배방정식에 대하여 10000개의 데이터를 추출하여 사용하였다. 경계조건을 보다 잘 만족하도록 하기 위하여, L_B 에 100을 곱하여 학습을 수행하였다.

다음으로 부정정 구조인 연속보 문제를 해결하기 위하여 Piecewise PINN를 적용하였다. 총 5000번의 에폭 동안 학습을 진행하였으며 다른 모든 학습 조건은 일반 PINN와 동일하게 설정하였다. 소영역 사이의 경계에서 100개씩의 데이터를 추출하여 학습에 사용하였다.

모든 PINN 모델은 Python 3.9.7 상에서 Pytorch 1.9.1, CUDA 10.2, cuDNN 7.6.5, Numpy 1.20.3을 사용하여 작성되었으며, Nvidia Quadro RTX 5000을 사용하여 학습되었다. 1차원 보에 대한 근사해의 정확도를 평가하기 위하여, COMSOL Multiphysics를 이용하여 얻은 FEM 기반의 결과와 PINN을 통해 얻은 근사해 사이의 결과를 비교하였다.

4.1.2 단순보 (Simple Beam) 문제의 해석

본 연구에서 해석하고자 한 단순보 문제는 다음과 같다 (Fig. 6; Eq.

8). 지배방정식은 앞서 언급하였던 1차원 보의 처짐에 관련된 4차 미분 방정식을 사용하였고, 보의 양 끝점은 각각 $x = -1$ 과 $x = 1$ 이고 두 점 모두에서 지지를 받아 처짐과 휨 모멘트가 0이라고 설정하였다. 하중은 2개의 불균일한 분포하중을 설정하였다. 보 재료의 단면 2차모멘트와 탄성계수는 각각 1 m^4 , 1 Pa 이라고 가정하였다.

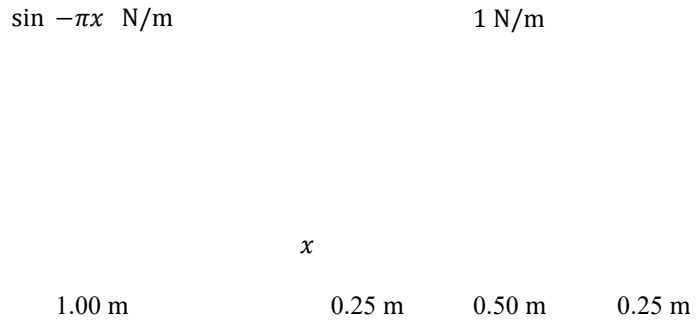


Fig. 6 Simple beam under two different distributed load

	$\frac{\partial^4 u}{\partial x^4} - f = 0,$ $u(-1) = u(1) = 0, \quad u_{xx}(-1) = u_{xx}(1) = 0$	(35)
--	---	------

where, $u(a)$ is a displacement of the beam at $x = a$, f is a composite load, and u_{xx} denotes $\frac{\partial^2 u}{\partial x^2}$

단순보 문제의 PINN을 통해 얻은 해석 결과와 FEM을 통해 얻은 해석 결과를 Fig. 7에 나타내었다. 학습을 마친 후의 손실은 0.0996으로 나타났다으며, FEM과 결과를 비교했을 때 거의 차이가 없는 것으로 판단되었다. 또한 학습 과정 중 경계조건에 대한 손실, 지배방정식에 대한 손실 및 총 손실을 각각 Fig. 8에 도시하였다. 학습의 극초반 과정에서는 경계조건에 대한 손실이 지배방정식에 대한 손실보다 훨씬 크게 나타났지만, 이후로는 정반대의 양상이 나타나 총 손실의 대부분을 지배방정식에 대한 손실이 차지하였다.

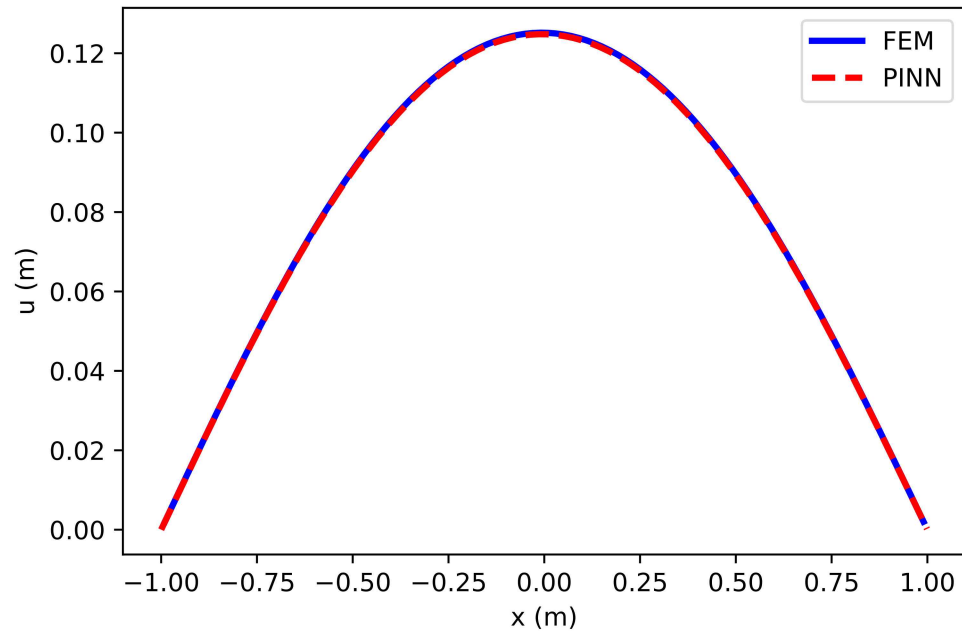


Fig. 7 A comparison between the PINN and FEM solutions of 1-D simple beam problem

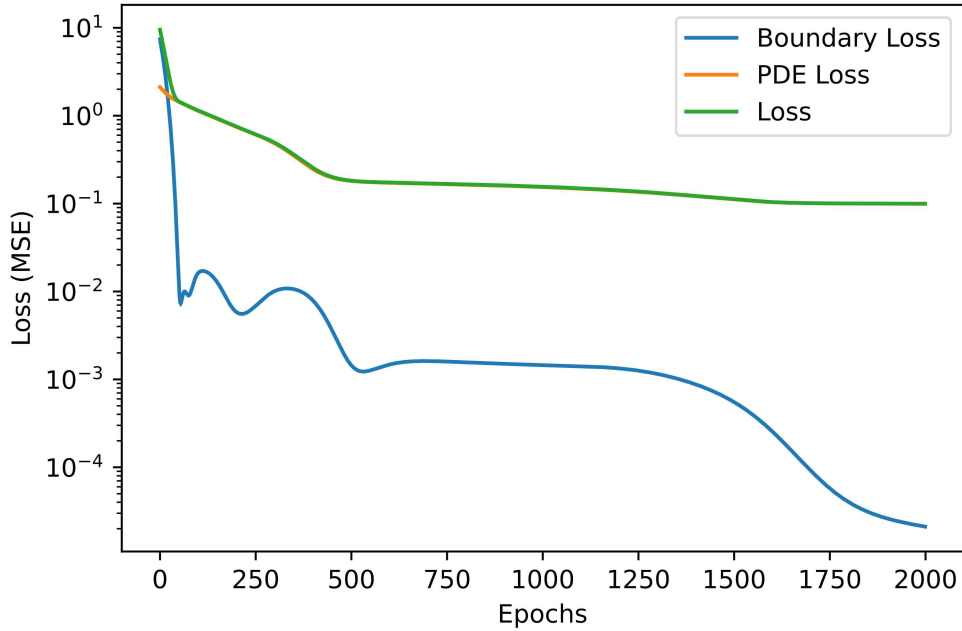


Fig. 8 A history of boundary, PDE, and total loss during training for 1-D simple beam problem

4.1.3 캔틸리버보 (Cantilever Beam) 문제의 해석

본 연구에서 해석하고자 한 캔틸리버보 문제는 다음과 같다. (Fig. 9; Eq. 9) 지배방정식은 마찬가지로 1차원 보의 처짐에 관련된 4차 미분방정식을 사용하였고, 보의 양 끝점은 각각 $x=-1$ 과 $x=1$ 이고 왼쪽 끝 점에서 지지를 받는다. 왼쪽 끝 점에서는 처짐과 처짐각이 0, 오른쪽 끝 점에서는 모멘트와 전단력이 0이라고 설정하였다. 하중은 2개의 불균일한 분포하중을 설정하였다. 보 재료의 단면 2차모멘트와 탄성계수는 각각 1 m^4 , 1 Pa 이라고 가정하였다.

캔틸리버 보 문제의 PINN을 통해 얻은 해석 결과와 FEM을 통해 얻은 해석 결과를 Fig. 7에 나타내었다. 학습을 마친 후의 손실은 0.0996으로 나타났으며, FEM과 결과를 비교했을 때 거의 차이가 없는 것으로 판단되었다. 또한 학습 과정 중 경계조건에 대한 손실, 지배방정식에 대한 손실 및 총 손실을 각각 Fig. 8에 도시하였다. 학습의 전 과정에서

지배방정식에 대한 손실이 경계조건에 대한 손실보다 크게 나타나, 총 손실의 대부분을 지배방정식에 대한 손실이 차지하였다. 또한 1000 에폭 이후로 경계조건에 대한 손실에서 지속적인 진동이 발생하였다.

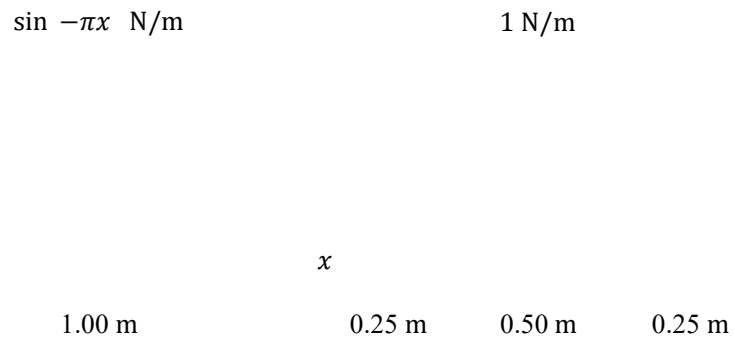


Fig. 9 Cantilever beam under two different distributed load

	$\frac{\partial^4 u}{\partial x^4} - f = 0,$ $u(-1) = u(1) = 0, \quad u_{xx}(-1) = u_{xx}(1) = 0$	(36)
--	---	------

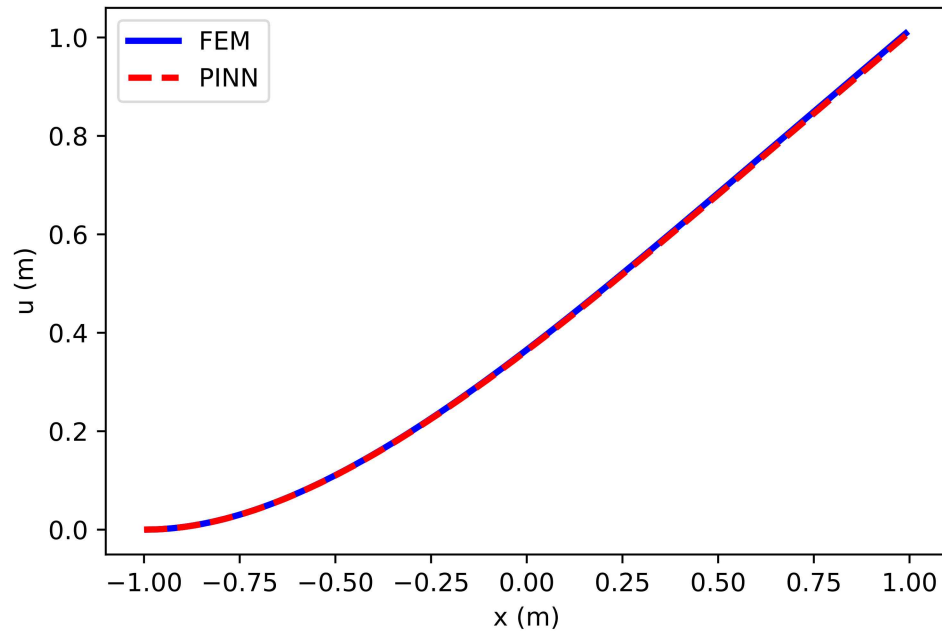


Fig. 10 A comparison between the PINN and FEM solutions of 1-D cantilever beam problem

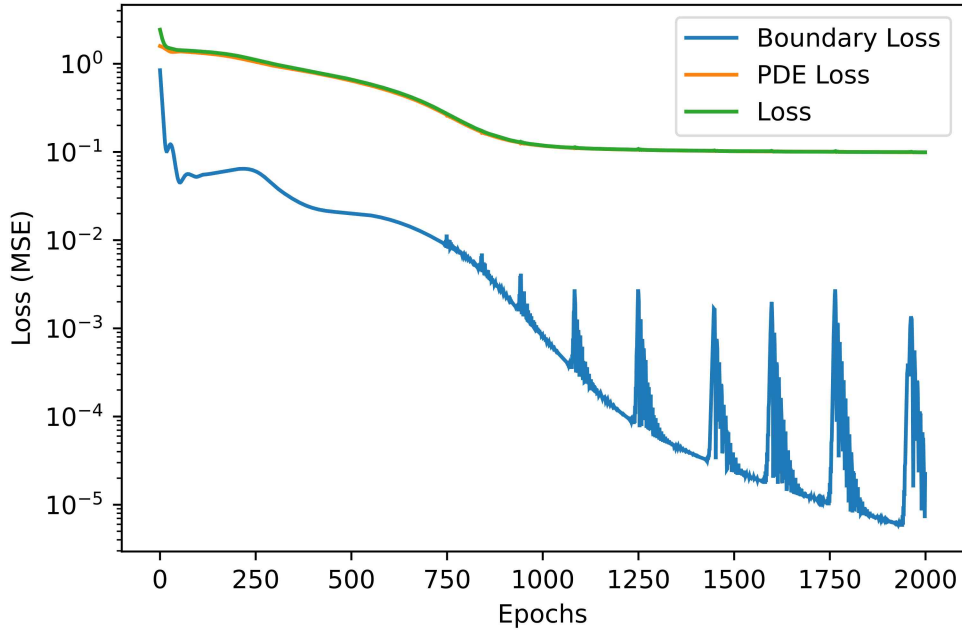


Fig. 11 A history of boundary, PDE, and total loss during training for 1-D cantilever beam problem

4.1.4 연속보 (Continuous Beam) 문제의 해석

본 연구에서 해석하고자 한 연속보 문제는 다음과 같다. (Fig. 12; Eq. 10) 지배방정식은 앞서와 마찬가지로 1차원 보의 처짐에 관련된 4차 미분방정식을 사용하였다. 보의 양 끝점은 각각 $x = -1$ 과 $x = 1$ 이고 추가적으로 $x = 0$ 에서 지지를 받는다는 조건을 추가하였다. 양 끝점에서는 처짐과 모멘트가 0이라고 설정하였고 중간 지지점에서는 처짐만이 0이라고 설정하였다. 하중은 1개의 불균일한 분포하중과 1개의 집중하중을 설정하였으며, 분포하중은 보 전체에 작용하고 집중하중은 $x = 0.5$ 에서 작용한다고 설정하였다. 마찬가지로 보 재료의 단면 2차모멘트와 탄성계수는 각각 1 m^4 , 1 Pa 이라고 가정하였다.

$$\cos -\frac{\pi}{2} x \text{ N/m} \quad 1 \text{ N}$$

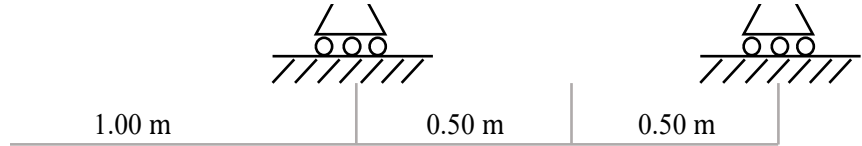


Fig. 12 Continuous beam under a distributed load and a point load

$\frac{\partial^4 u}{\partial x^4} - f = 0,$ $u(-1) = u(0) = u(1) = 0, \quad u_{xx}(-1) = u_{xx}(1) = 0$	(37)
--	------

연속보 문제는 전체 연산 영역에 대하여 한 개의 해를 구하는 일반적인 PINN과 연산 영역을 소영역으로 분할하여 각 소영역의 해를 구하는 piecewise PINN의 2가지 방법을 사용하여 해석하였다. 이때 영역의 분할 결과는 Fig. 9와 같았다. 우선 $x=0.5$ 에 집중하중이 가해지기 때문에 그 좌우에서 전단력의 불연속을 구현하기 위하여 해당 지점에서 영역을 분할하였고, 또 $x=0$ 에서 추가적으로 영역을 분할하여 모든 소영역이 정정 문제가 되도록 하였다.

해석 결과를 비교해보면, 일반 PINN은 연속보 문제를 잘 해석하지 못하는 것으로 드러났으며, piecewise PINN을 통해 얻은 해는 FEM을 통해 얻은 해와 거의 차이가 없음이 드러났다 (Fig. 14). 이를 통해 영역을 소영역으로 분할하지 않을 시 부정정 문제를 해결하지 못할뿐더러, 집중하중과 같은 조건도 구현하지 못함을 알 수 있었다.

학습 중 손실의 변화를 살펴보면, 3개의 소영역 모두에서 L_I 가 초반에 가장 크게 나타났으나 학습이 진행됨에 따라 앞서 서술한 일반 PINN에서의 학습처럼 총 손실에서 차지하는 비중은 L_F 가 가장 컸고 그

다음으로 L_I , L_B 순으로 크게 나타났다. 3개의 소영역에서 학습 중 손실의 변화를 비교해보면 1번 소영역의 손실이 가장 작았으며 2, 3번 소영역의 손실 변화는 유사한 양상을 보였다.

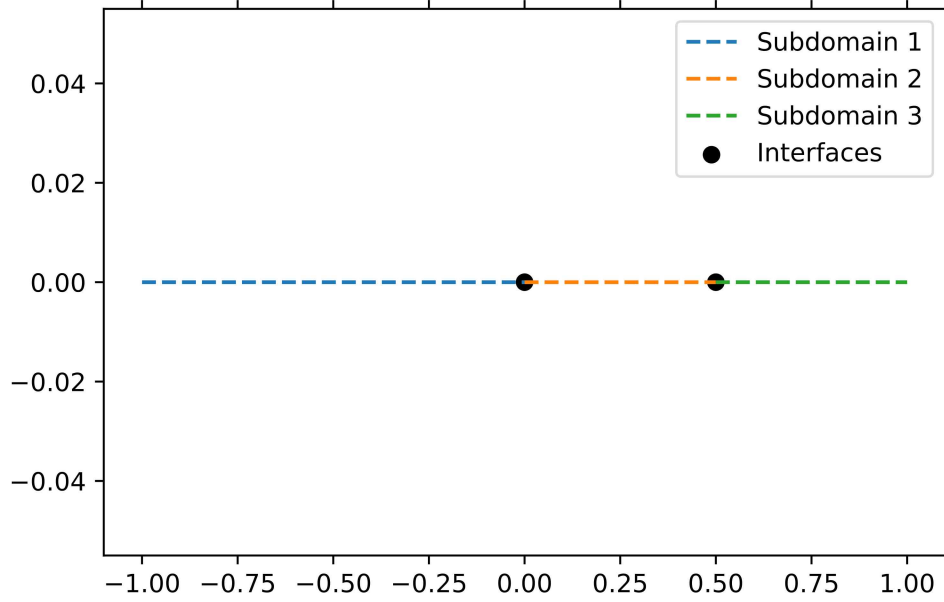


Fig. 13 Domain and interfaces decomposed to solve the 1-D continuous beam problem using piecewise PINN

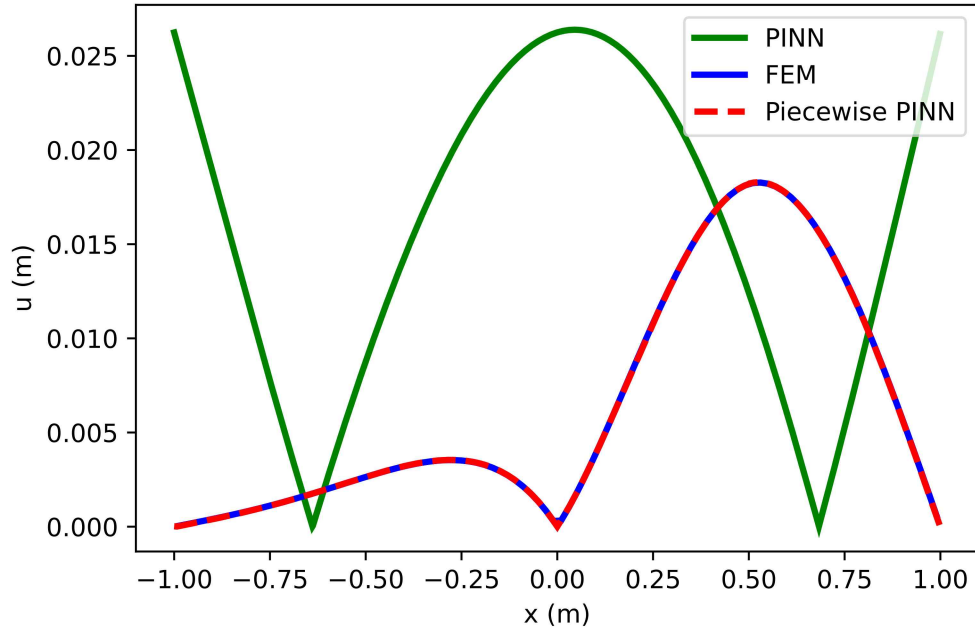


Fig. 14 A comparison among the PINN, piecewise PINN and FEM solutions of 1-D continuous beam problem

(a)

(b)

(c)

(d)

Fig. 15 A history of loss during training for 1-D continuous beam problem:

(a), (b), (c) are the history of boundary, PDE, interface, and total loss during training submodel 1, 2, 3, respectively. (d) is a comparison among the total loss of each submodel during training.

4.2 2차원 평면 응력 문제의 해석

4.2.1 문제의 설정

4.2.2 정정 평면 응력 문제의 해석

앞서 3.2에서 서술하였던 지배방정식을 사용하여 2차원 평면 응력 문제에서 x 방향 처짐과 y 방향 처짐을 모의하였으며, 해석하고자 한 문제는 다음과 같다 (Fig.). 이때, 체적력 (Body Force)는 다음과 같이 정의하였다 (Eq.). 또한 포아송 비 ν 와 탄성계수 E 는 각각 0과 2 Pa로 설정하였다.

	$f_x = -10y \cos\left(\frac{\pi}{2}y\right) \sin(\pi x),$ $f_y = -10x \sin\left(\frac{\pi}{2}y\right) \cos(\pi x)$	(38)
--	--	------

PINN으로 2차원 평면 응력 문제를 해석한 결과와 FEM으로 같은 문제를 해석한 결과를 Fig. 16에 도시하였다. 학습 결과 최종적으로 손실은 0.038이었으며, 해석 결과는 u , v 모두 FEM과 비교해서 큰 차이가 없는 것으로 드러났다.

학습 중 손실의 변화를 Fig. 15에 제시하였다.

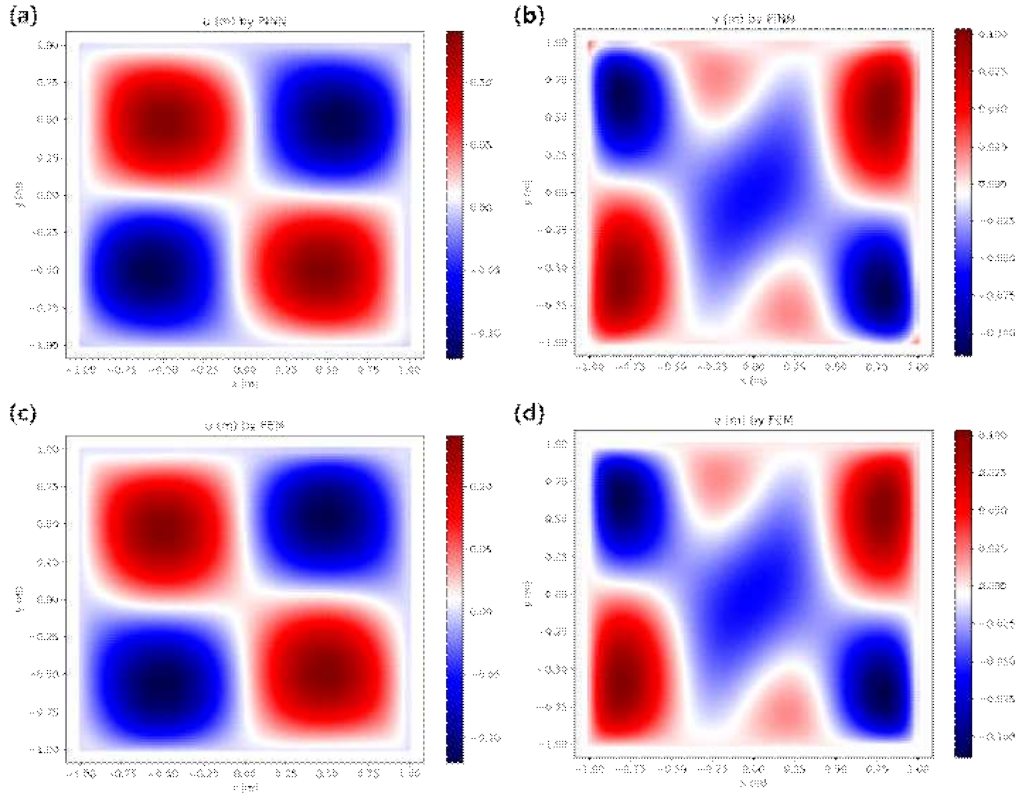


Fig. 16 A comparison between the PINN and FEM solutions of 2-D plane-stress problem: (a) and (b) are the X and Y component of displacement predicted by PINN; (c) and (d) are the X and Y component of displacement calculated by FEM.

4.2.3 부정정 평면 응력 문제의 해석

4.3 2차원 판 문제의 해석

4.3.1 문제의 설정

4.3.2 정정 판 문제의 해석

4.3.3 부정정 판 문제의 해석

4.4 PINN과 piecewise PINN의 성능 비교

제 5 장 요약 및 결론

제 6 장 참고문헌

제 7 장 Abstract

참 고 문 헌(16pt)

Abstract(16pt)

외 국 어 논 문 제 목(22pt)

성 명(영문명)(14pt)

학과 및 전공(영문명)

The Graduate School

Seoul National University

논문의 내용과 결론에 관하여 간략하고 구체적으로 기재(11pt)

keywords : (6단어 이내)

Student Number :