

# PINN Hot Papers in Hemodynamics and Methodology

2025년 후반 – 2026년 5월 동향 보고서

*with Lang-PINN deep-dive and multi-domain extensions*

**Compiled for:** Crowdy (CTO, PlanitAI)

**Compiled by:** Claude (Anthropic) – Opus 4.7

**Date:** 2026년 5월 20일

**Scope:** 대화 전체 + Deep Research 원본

**Class:** 내부 자료 (Internal)

## About this document

이 문서는 Crowdy와 Claude의 대화 세션(2026년 5월 20일)에서 진행된 PINN 관련 리서치를 LaTeX로 정리한 것입니다. 본문은 다음 5개 챕터로 구성되어 있습니다.

- Chapter 1: 대화 흐름 요약 (Conversation Timeline)
- Chapter 2: PINN 핫 페이퍼 종합 리포트 (Deep Research Output)
- Chapter 3: Lang-Pinn 첫 번째 해설
- Chapter 4: Lang-Pinn 심화 해설 – Architecture, Math, Critique
- Chapter 5: Lang-Pinn의 혈류 외 응용 도메인

# 목차

<b>1 대화 흐름 요약 (Conversation Timeline)</b>	<b>4</b>
1.1 Step 0 – 초기 질문	4
1.2 Step 1 – 범위 확정	4
1.3 Step 2 – Deep Research 결과 산출	5
1.4 Step 3 – Lang-Pinn 후속 질문	5
1.5 Step 4 – Lang-Pinn 심화 요청	5
1.6 Step 5 – 도메인 확장 탐색	5
1.7 Step 6 – 본 문서 컴파일 요청	6
<b>2 PINN 핫 페이퍼 종합 리포트</b>	<b>7</b>
2.1 TL;DR	7
2.2 Key Findings	8
2.3 Tier 1: 혈류·심혈관 PINN (Late 2025 – May 2026)	8
2.3.1 PINGS-X – Physics-Informed Normalized Gaussian Splatting for 4D Flow MRI Super-Resolution (AAAI 2026)	8
2.3.2 Lannelongue et al. – Physics-Constrained GNN for Real-Time Intracranial Aneurysm Hemodynamics	9
2.3.3 PUNCH – Physics-Informed Uncertainty-Aware Network for Coronary Hemodynamics	9
2.3.4 DCP-INN – Dual-Correction PINN for Hemodynamic Reconstruction from Sparse Data	10
2.3.5 SinoFlow – CT Sinogram-Based PINN for Cardiovascular Flow Estimation	10
2.3.6 Imaging-Derived Coronary FFR / PINN-PINO 리뷰	10
2.3.7 Hematocrit-Dependent Rheology PINN for 4D-Flow MRI	11
2.3.8 Fast Inverse Blood Flow via PINNs (EPFL LHTC)	11
2.3.9 Physics-Informed Neural Operators for Cardiac Electrophysiology	11
2.3.10 Physics-Informed Deep Learning Surrogates for AAA (PINN + DeepONet)	11
2.3.11 Physics-Informed GNN for Carotid Artery Flow Field Estimation	11
2.3.12 Hybrid CFD-PINN-FSI for Coronary Trees (FFR/WSS)	12
2.4 Tier 2: 일반 PINN 방법론 핫 페이퍼 (Late 2025 – May 2026)	12

- 2.4.1 Gradient Alignment via Quasi-Second-Order Optimization (SOAP) — NeurIPS 2025 . . . . . 12
- 2.4.2 PIKAN vs PINN Unified Benchmark . . . . . 12
- 2.4.3 Lang-Pinn — LLM Multi-Agent PINN Builder (ICLR 2026 submission) 12
- 2.4.4 AB-PINNs — Adaptive-Basis Residual-Driven Domain Decomposition 13
- 2.4.5 Architectural Physics Embedding for Wave Field Reconstruction . 13
- 2.4.6 P-PINN — Selective Pruning for Noisy Inverse Problems . . . . . 13
- 2.4.7 Diffusion Models with Physics-Guided Inference . . . . . 13
- 2.4.8 Rethinking Input Domains via Geometric Compactification Mappings . . . . . 13
- 2.4.9 HyResPINNs — Hybrid Residual PINN with Adaptive Function Spaces 14
- 2.4.10 Learnable Loss Balancing + Transfer Learning PINN (ICLR 2026 submission) . . . . . 14
- 2.4.11 Training Deep Physics-Informed Kolmogorov-Arnold Networks . . 14
- 2.4.12 Saddle-Point Reformulation for PINN Stability . . . . . 14
- 2.5 관찰된 트렌드 . . . . . 14
- 2.6 Recommendations (CTO 관점) . . . . . 15
- 2.7 Caveats . . . . . 15
- 3 Lang-PINN 첫 번째 해설 . . . . . 16**
- 3.1 왜 이 논문이 중요한가 — 해결하려는 문제 . . . . . 16
- 3.2 4-Agent 아키텍처 . . . . . 16
  - 3.2.1 Agent 1: PDE Agent — Linguistic Variability를 격파하는 방법 . . . . . 16
  - 3.2.2 Agent 2: PINN Agent — Training-Free Architecture Selection . . . . . 17
  - 3.2.3 Agent 3: Code Agent — Modular Generation + PDE Loss Verification 17
  - 3.2.4 Agent 4: Feedback Agent — Closed Loop의 핵심 . . . . . 17
- 3.3 실험 결과 정리 . . . . . 18
- 3.4 한계와 비판적 평가 . . . . . 18
- 3.5 CTO 관점에서의 인사이트 . . . . . 18
- 4 Lang-PINN 심화 해설 — Architecture, Math, Critique . . . . . 20**
- 4.1 논문의 출발점 — 세 가지 통제 실험 결과 . . . . . 20
  - 4.1.1 실험 ① — Linguistic Variability of Task Formulation . . . . . 20
  - 4.1.2 실험 ② — Variability of Architecture Performance Across PDEs . . . 20
  - 4.1.3 실험 ③ — Complexity of Code Generation . . . . . 20
- 4.2 4-Agent 시스템의 수학적 정밀화 . . . . . 21
  - 4.2.1 PDE Agent — 두 개의 직교적 유사도 척도 . . . . . 21
  - 4.2.2 PINN Agent — Bradley-Terry로 architecture 능력 측정 . . . . . 21
  - 4.2.3 Code Agent — Round-Trip Symbolic Verification . . . . . 22
  - 4.2.4 Feedback Agent — 4-Dimensional Quality Score . . . . . 22
- 4.3 Task2PDE 벤치마크 — 구체 예시 . . . . . 23
  - 4.3.1 Level 1 (clean, paper-style) . . . . . 23
  - 4.3.2 Level 2 (irrelevant but realistic side information) . . . . . 23

4.3.3	Level 3 (ambiguous shorthand)	23
4.3.4	Level 4 (disorganized lab notebook)	24
4.4	실험 결과 — Per-PDE Breakdown	24
4.5	Sibling Systems와의 비교	24
4.6	Critical Assessment	25
4.6.1	강점	25
4.6.2	약점·한계	25
4.7	Multi-Agent System Design Pattern으로서의 가치	25
<b>5</b>	<b>Lang-PINN의 혈류 외 응용 도메인</b>	<b>27</b>
5.1	Tier 1 — 즉시 적용 가능 (Lang-Pinn의 sweet spot)	27
5.1.1	열전달·전자기기 thermal management	27
5.1.2	음향·진동·구조 동역학	27
5.1.3	화학 반응 시스템·반응-확산	27
5.2	Tier 2 — 적용 가능하나 약간의 확장 필요	28
5.2.1	지하수·환경 유체 (porous media flow)	28
5.2.2	옵션 가격결정·금융 PDE (Black-Scholes 계열)	28
5.2.3	광학·포토닉스 (전자기 wave propagation)	28
5.3	Tier 3 — 흥미롭지만 도전적 영역	29
5.3.1	양자역학 — Schrödinger equation	29
5.3.2	우주물리·천체역학 — General Relativity	29
5.3.3	인구역학·역학 (epidemiology, ecology)	29
5.3.4	재료과학 — Phase field method	29
5.4	CTO 관점 — 어디부터 시작할까	29
<b>A</b>	<b>컴파일 노트</b>	<b>31</b>
A.1	빌드 방법	31
A.2	소스 메타데이터	31
A.3	원본 대화 메타데이터	32
A.4	이 문서의 한계	32

# Chapter 1

## 대화 흐름 요약 (Conversation Timeline)

이 챗터는 본 리서치 세션에서 Crowdy(사용자)와 Claude(AI assistant) 사이에 오간 질문과 답변의 흐름을 시간 순으로 정리한 것입니다. 각 단계의 의도와 결과가 후속 챗터로 어떻게 연결되는지 보여주는 메타 레벨 트레이스(meta-level trace)입니다.

### 1.1 Step 0 — 초기 질문

**Crowdy:** 올해 나온 Pinn 논문중 가장 핫한 논문 리스트를 알려줘. 나는 특히 혈류분석에 Pinn을 사용하는 방법에 관심이 있어.

Claude는 본격적인 리서치 전에 범위를 좁히기 위해 세 가지를 확인했습니다:

1. “올해”의 범위 — 2026년만 vs. 2025년 후반 포함
2. 혈류분석의 초점 영역 (뇌동맥류, 관상동맥, 대동맥 등)
3. 논문 성격 (방법론적 신규성 vs. 임상 응용)

### 1.2 Step 1 — 범위 확정

**Crowdy:**

1. 2025년 말부터
2. 모두
3. 방법론적 신규성

이 답변을 바탕으로 Claude는 extended search task를 실행했습니다. 검색 대상은 arXiv (cs.LG, physics.flu-dyn, physics.med-ph, q-bio.QM), Google Scholar, NeurIPS 2025 / ICLR 2026 / AAAI 2026, 그리고 주요 저널들(JCP, CMAME, Nature Machine Intelligence, Comput-

ers in Biology and Medicine, Medical Image Analysis, IEEE TMI, Computers & Fluids)이었습니다.

### 1.3 Step 2 — Deep Research 결과 산출

리서치 결과는 Chapter 2에 전체 수록되어 있습니다. 주요 발견 사항:

- Tier 1 (혈류 PINN) 12편 식별
- Tier 2 (일반 방법론) 12편 식별
- 결정적 3편: PINGS-X (AAAI 2026), Lannelongue et al. (npj Digital Medicine), PUNCH (arXiv:2601.17192)
- 방법론 핵심 흐름: 2차 최적화/SOAP, PIKAN 정착, LLM 멀티에이전트, 동적 도메인 분해

### 1.4 Step 3 — Lang-Pinn 후속 질문

**Crowdy:** “LLM 에이전트가 PINN 개발 루프에 진입 (Lang-PINN...)” — 이 내용을 좀 더 자세히 설명해줘.

Claude는 arXiv:2510.05158을 fetch하고 4-agent 아키텍처의 동작 원리를 정리했습니다. Chapter 3에 수록.

### 1.5 Step 4 — Lang-Pinn 심화 요청

**Crowdy:** Lang-Pinn에 대해 좀 더 자세하게 설명해줘.

Claude는 OpenReview에서 논문 PDF 전체를 fetch하여 sibling system 비교, Bradley-Terry 수학적 디테일, Task2PDE의 구체 예시, 14개 PDE per-PDE 결과, multi-agent design pattern 6개까지 심화 분석을 수행했습니다. Chapter 4에 수록.

### 1.6 Step 5 — 도메인 확장 탐색

**Crowdy:** Lang-Pinn을 혈류 시뮬레이션 CFD 이외의 다른 영역에서 사용한다면 어떤 영역을 예로 들 수 있을까?

Claude는 PINNacle 호환성과 일본 산업 적합성을 기준으로 10개 도메인을 3-tier로 분류했습니다. Chapter 5에 수록.

## 1.7 Step 6 — 본 문서 컴파일 요청

마지막 단계로 Crowdy가 전체 대화와 리서치 결과를 LaTeX 파일로 컴파일해서 외부 서버에 업로드 요청. Claude는 외부 도메인 접근 제약과 보안 우려를 설명하고 로컬 파일 생성 + 다운로드 옵션을 제시. 옵션 선택 결과:

- Document class: **report**
- Scope: **대화 전체 + research 원본**
- Language: **한국어 우선 (kotex, 나눔글꼴)**

이 문서가 그 산출물입니다.

# Chapter 2

## PINN 핫 페이퍼 종합 리포트

이 챕터는 Deep Research가 산출한 원본 리포트 전체입니다. 형식과 인용은 원본을 유지하며, 한국어 표기와 약간의 조판만 조정했습니다.

### 2.1 TL;DR

- **혈류(Tier 1)의 결정적 3편:** 이 기간의 가장 주목할 Pinn 혈류 논문은
  1. **PINGS-X** (Hanyang University, AAI 2026) — 4D Flow MRI 초해상화에서 환자별 Pinn 학습 병목을 axes-aligned Gaussian Splatting 표현으로 대체
  2. **Lannelongue et al.** (MINES Paris/PSL, npj Digital Medicine 9:212, 06 Feb 2026) — 물리 제약 GNN으로 두개내동맥류 혈역학을 실시간(GPU 1장에서 약 1분/심장박동주기) 예측하고 BenchAnXplore (105 환자 geometry) 벤치마크 공개
  3. **PUNCH** (AngioInsight + UCLA + U. Michigan + UC Riverside, [arXiv:2601.17192](https://arxiv.org/abs/2601.17192), v2 Feb 2026) — variational inference 결합 Pinn으로 표준 관상동맥 조영술만으로 CFR을 비침습 추정(GPU 1장에서 약 3분/환자)
- **방법론(Tier 2) 핵심 흐름:** 2025-2026년 Pinn 방법론은
  1. **2차 최적화/그래디언트 정렬** (NeurIPS 2025의 Sifan Wang-Paris Perdikaris, SOAP optimizer로 Re=10,000 난류 최초 달성)
  2. **PIKAN(KAN backbone) 정착** (Dzimah et al. [arXiv:2602.15068](https://arxiv.org/abs/2602.15068), 통제 실험에서 MLP-Pinn 능가)
  3. **LLM 멀티에이전트 자동화** (Lang-Pinn, ICLR 2026 submission)
  4. **동적 도메인 분해** (AB-PINNs, MERL+Cornell)로 수렴
- **CTO 의사결정 권고:** 임상/제품화는 PUNCH·Lannelongue·PINGS-X 라인이 가장 빠른 경로(분~10분 단위 추론, 환자별 retraining 불필요). 내부 PDE 솔버 인프라는 **SOAP 오티마이저** + **AB-PINN 동적 도메인 분해** + **PIKAN 백본**의 3축 표준화가 가장 위험-보상비가 우수.

## 2.2 Key Findings

1. **혈류 Pinn은 “환자별 재학습” 병목을 깨는 방향으로 급격히 이동.** 전통 Pinn은 환자당 수 시간 학습이 필요했으나, 2025년 말부터 (a) 명시적 표현(PINGS-X의 4D Gaussian Splatting), (b) 그래프 신경 연산자(Lannelongue GNN), (c) Pinn/PINO 결합(Coronary FFR 리뷰 by Tanxin Zhu·Weihua Zhou)이 임상 시계열(분 단위)에 도달.
2. **Inverse 혈류 문제에서 불확실성 정량화(UQ)가 표준이 되는 중.** PUNCH가 KL-regularized variational posterior와 100회 Monte Carlo 샘플로 신뢰구간을 산출하는 것은, FDA·CE 인증의 가장 큰 장벽인 “calibration & gatekeeping”을 정면 해결한 사례.
3. **2026년 초 Pinn 학습 이론의 결정적 진전은 그래디언트 충돌 해소.** Sifan Wang(Yale Institution for Foundation of Data Science) + Ananyae Kumar Bhartari + Paris Perdikaris(UPenn MEAM)의 NeurIPS 2025 SOAP 결과는, 수년간 정체되어 있던 “PINN은 turbulent flow를 못 푼다”는 한계를 처음으로  $Re=10^4$ 까지 해소.
4. **PIKAN(Physics-Informed Kolmogorov-Arnold Networks)이 MLP-Pinn 대비 우위가 통제 실험으로 검증됨** (Dzimah et al., MIT+UCM, [arXiv:2602.15068](https://arxiv.org/abs/2602.15068), Feb 14, 2026). 동일 파라미터 예산에서 PIKAN이 더 높은 정확도·더 적은 iteration·더 좋은 gradient 추정.
5. **LLM 에이전트가 Pinn 개발 루프에 진입** (Lang-Pinn, [arXiv:2510.05158](https://arxiv.org/abs/2510.05158), ICLR 2026 submission): 자연어 → PDE → 아키텍처 → 코드 → 디버깅의 4-에이전트 파이프라인.
6. **CT 기반 혈류 추정의 패러다임이 “이미지 후처리”에서 “raw sinogram 직접 학습”으로 전환** (SinoFlow, UCSD Bioengineering, [arXiv:2511.03876](https://arxiv.org/abs/2511.03876)): 시간 의존 Radon 변환을 Pinn 손실에 내장.
7. **Architectural physics embedding의 부상:** soft loss만이 아니라 fundamental solution이나 integrator를 아키텍처에 하드 임베딩 (Zhang-Ye-Ma의 PE-Pinn, [arXiv:2603.02231](https://arxiv.org/abs/2603.02231); Point Neuron Learning; HRPINN).

## 2.3 Tier 1: 혈류·심혈관 PINN (Late 2025 – May 2026)

### 2.3.1 PINGS-X — Physics-Informed Normalized Gaussian Splatting for 4D Flow MRI Super-Resolution (AAAI 2026)

**arXiv** [arXiv:2511.11048](https://arxiv.org/abs/2511.11048) (Nov 14, 2025; revised Jan 13, 2026)

**Authors / Institution** Sun Jo, Seok Young Hong, Jinhyun Kim, Seungmin Kang, Ahjin Choi, Don-Gwan An, Simon Song, Je Hyeong Hong — 주로 **Hanyang University (서울)**, 일부 NTU Singapore.

**혁신** 기존 Pinn 기반 4D Flow MRI 초해상화의 핵심 단점은 환자별로 Pinn을 재학습해야 한다는 점. PINGS-X는 3D Gaussian Splatting(3DGS)에서 영감을 얻어 시공간을 **axes-aligned 4D 정규 Gaussian 집합**으로 명시적으로 표현하고, normalization + 축 정렬 제약 + Navier-Stokes 일관성 손실을 결합. 결과: 합성/실측 4D Flow MRI 양쪽에서 Pinn-SR 베이스라인 대비 학습 시간이 크게 단축되면서 carotid artery의 fine-scale 유동 재현 향상.

**임상 의의** 협착/동맥류 위험 지표(WSS, vorticity) 추출용 high-res 4D Flow를 짧은 스캔 시간으로 확보, 임상 워크플로에 통합 가능.

**Link** <https://arxiv.org/abs/2511.11048>

### 2.3.2 Lanelongue et al. — Physics-Constrained GNN for Real-Time Intracranial Aneurysm Hemodynamics

**DOI** 10.1038/s41746-026-02404-z, **npj Digital Medicine** vol. 9, article no. 212, published 06 February 2026

**Authors / Institution** Vincent Lanelongue, Paul Garnier, Pablo Jeken-Rico, Aurèle Goetz, Philippe Meliga, Yves Chau, Elie Hachem — **MINES Paris, PSL University, CEMEF**

**혁신** 환자별 두개내동맥류 혈역학(속도장·WSS·OSI)을 그래프 신경망 위에 Navier-Stokes 기반 물리 제약 손실을 부과해 학습. “**BenchAnXplore**” — 105 환자 동맥류 geometry와 CFD 정답 필드의 대규모 벤치마크 데이터셋 공개 (per follow-up [arXiv:2512.09013](https://arxiv.org/abs/2512.09013)). CFD 대비 약  $200\times$  속도(64 CPU 시간 단위 → GPU 1장에서 약 1분/심장박동주기).

**임상 의의** 두개내동맥류 파열 위험 평가(SAWSS 등)를 실시간으로 환자별 산출 가능. 수술적 결정(coil/stent) in-loop 활용 가능성.

**Link** <https://www.nature.com/articles/s41746-026-02404-z>

**Follow-up** 동일 그룹의 [arXiv:2512.09013](https://arxiv.org/abs/2512.09013) — 51M 파라미터 transformer-GNN, 약 28.5억 node-timestep 학습; 전체 파이프라인(segmentation→meshing→inference) 10분 미만.

### 2.3.3 PUNCH — Physics-Informed Uncertainty-Aware Network for Coronary Hemodynamics

**arXiv** [arXiv:2601.17192](https://arxiv.org/abs/2601.17192) (Jan 2026; v2 Feb 7, 2026)

**Authors / Institution** Sukirt Thakur (**AngioInsight Inc., lead**), Marcus Roper (UCLA), Yang Zhou, Dmitry Yu. Isaev, Reza Akbarian Bafghi (CU Boulder), Brahmajee K. Nallamothu (U. Michigan), C. Alberto Figueroa (U. Michigan), Srinivas Paruchuri, Scott Burger, Carlos Collet (Cardiovascular Research Foundation, NY), **Maziar Raissi (UC Riverside)**

**혁신** 표준 관상동맥 조영술 영상만으로 coronary flow reserve(CFR)을 비침습 추정 — invasive pressure wire 불요. Pinn으로 조영제(contrast) 이동의 first-principles 방정식(intravascular tracer dynamics)을 풀고, **variational inference** (posterior  $q_\phi(z) = \mathcal{N}(\mu, \text{diag}(\sigma^2))$ )를 표준정규분포에 KL 발산으로 정규화)를 결합. 100회 Monte Carlo 샘플로 CFR 신뢰구간 산출. 환자당 population-level 학습 불필요. 합성 조영술 + 환자 20명 bolus thermodilution 데이터로 검증.

**임상 의의** 허혈성 심장 질환 평가 환자의 약 70%에서 obstructive lesion이 발견되지 않으며 그 절반이 미진단 coronary microvascular dysfunction(CMD). PUNCH는 routine 조영술

데이터에서 CMD를 정량화하는 첫 Pinn 솔루션. 여성 환자군의 진단 격차 해소 가능성.

**Link** <https://arxiv.org/abs/2601.17192>

### 2.3.4 DCP-INN — Dual-Correction PINN for Hemodynamic Reconstruction from Sparse Data

**arXiv** [arXiv:2605.12544](https://arxiv.org/abs/2605.12544) (May 9, 2026)

**Authors** Jingtai Song, Qinsheng Zhu, Xiaodong Xing, Yufeng Tang, Zhiyun Zhang, Xi-anwen Zhang, Hao Wu (중국 기반 그룹; 추정 UESTC)

**혁신** 두개내 internal carotid artery의 고도로 굴곡진(tortuous) geometry에서 sparse 측정(transcranial Doppler / CTA)만으로 전체 유동장 복원. (1) **diamond-shaped main network**가 low-frequency 물리 추세를, (2) **wide-deep correction network**가 high-frequency 잔차를 병렬 보정. (3) **causal decoupling** 학습 전략 + 고차 Taylor 전개 기반 물리 손실로 local 연속성 강화.

**Link** <https://arxiv.org/abs/2605.12544>

### 2.3.5 SinoFlow — CT Sinogram-Based PINN for Cardiovascular Flow Estimation

**arXiv** [arXiv:2511.03876](https://arxiv.org/abs/2511.03876) (Nov 2025)

**Authors / Institution** Jinyuxuan Guo (UCSD Bioengineering), Gurnoor Singh Khurana (UCSD CSE), Alejandro Gonzalo Grande (UW Mech. Eng.), Juan C. del Alamo (UW), **Francisco Contijoch** (UCSD)

**혁신** Pinn을 재구성 이미지(FBP 결과)에 적용하던 ImageFlow 패러다임을 깨고, **CT의 forward acquisition 모델(시간 가변 Radon 변환)을 Pinn 손실 안에 직접 내장**. raw sinogram에서 직접 속도장 추정 → reconstruction artifact가 유동 추정을 오염시키는 문제 우회. Gantry rotation 속도·tube current·pulse mode 변동에 강건. 2D 이상화 vessel bifurcation에서 시뮬레이션 검증.

**Link** <https://arxiv.org/abs/2511.03876>

### 2.3.6 Imaging-Derived Coronary FFR / PINN-PINO 리뷰

**arXiv** [arXiv:2602.16000](https://arxiv.org/abs/2602.16000) (Feb 17, 2026)

**Authors / Institution** Tanxin Zhu (lead), Emran Hossen, Chen Zhao, Michele Esposito, Jiguang Sun, **Weihua Zhou (senior) — Michigan Technological University**

**내용** CT/조영술 기반 imaging-derived FFR에서 CFD → ML/DL → Pinn/PINO로의 진화를 종합. **deployment-oriented 지표(calibration, UQ, quality-control gatekeeping)**를 임상 적용의 필수 조건으로 명시.

**Link** <https://arxiv.org/abs/2602.16000>

### 2.3.7 Hematocrit-Dependent Rheology PINN for 4D-Flow MRI

arXiv [arXiv:2508.03326](https://arxiv.org/abs/2508.03326) (Aug 5, 2025)

**Authors / Institution** Moises Sierpe, Ernesto Castillo (**Universidad de Santiago de Chile**), Hernán Mella (PUC Valparaíso), **Felipe Galarce** (PUC Valparaíso, corresponding)

**혁신** 환자별 aortic geometry + 합성 4D-flow MRI에서 **shear-thinning 비뉴턴 유동 + hematocrit 의존성**을 Pinn으로 동시 추정. **curriculum training + adaptive collocation points + adaptive loss balancing**의 3중 안정화. **압력 강하 추정 상대 오차 약 1%**; vW-ERP와 결합해 super-sampled 고차 압력 추정.

**Link** <https://arxiv.org/abs/2508.03326>

### 2.3.8 Fast Inverse Blood Flow via PINNs (EPFL LHTC)

arXiv [arXiv:2604.03221](https://arxiv.org/abs/2604.03221) (Apr 2026)

**Institution** EPFL Laboratory of Hemodynamics and Cardiovascular Technology (LHTC)

**혁신** 1D arterial network에서 **단일 비침습 측정(brachial cuff pressure)** 만으로 8개 동맥 트리 전체를 단일 Pinn으로 inverse-solve. **Terminal resistance  $R_T$ 와 compliance  $C_T$** 를 학습 가능 파라미터로 추가; 환자별 5-10분 학습.

**Link** <https://arxiv.org/html/2604.03221>

### 2.3.9 Physics-Informed Neural Operators for Cardiac Electrophysiology

arXiv [arXiv:2511.08418](https://arxiv.org/abs/2511.08418) (Nov 2025; PMLR 2026 accepted)

**혁신** 심장 EP의 PDE를 Pinn 대신 **Fourier Neural Operator** + 물리 손실로 학습 — mesh-resolution 일반화 가능, 새 IC/parameter 재학습 불필요. 3D EP 시뮬레이션에서 Pinn 대비 일반화 향상. Aliev-Panfilov cell model 검증.

**Link** <https://arxiv.org/pdf/2511.08418>

### 2.3.10 Physics-Informed Deep Learning Surrogates for AAA (PINN + DeepONet)

HAL [hal-05191728](https://hal.science/hal-05191728) (late 2025/early 2026, Aix-Marseille A\*MIDEX 과제)

**혁신** AAA(복부대동맥류)에서 Pinn + DeepONet 결합. 3D Navier-Stokes 기반 Pinn을 환자별 학습 후, physics-informed DeepONet으로 새 boundary condition에 instant 적응.

**Link** <https://hal.science/hal-05191728/document>

### 2.3.11 Physics-Informed GNN for Carotid Artery Flow Field Estimation

arXiv [arXiv:2408.07110](https://arxiv.org/abs/2408.07110) (v2 Feb 20, 2026)

**Authors** Julian Suk et al. (Twente / UMC Utrecht)

**혁신** PointNet++ + SE(3)-steerable layers의 group-equivariant 그래프 네트워크. **4D Flow MRI 학습 데이터의 노이즈를 물리 손실로 정규화**하며 unseen carotid geometry에 일반화.

**Link** <https://arxiv.org/abs/2408.07110>

### 2.3.12 Hybrid CFD-PINN-FSI for Coronary Trees (FFR/WSS)

MDPI Fluids 9(12):280 (2024)와 2025 후속 thoracoabdominal aneurysm 연구. **혁신**: 1D Pinn으로 outlet boundary condition 계산 → 3D FSI에 주입하는 hybrid 파이프라인.

## 2.4 Tier 2: 일반 PINN 방법론 핫 페이퍼 (Late 2025 – May 2026)

### 2.4.1 Gradient Alignment via Quasi-Second-Order Optimization (SOAP) — NeurIPS 2025

**arXiv** [arXiv:2502.00604](https://arxiv.org/abs/2502.00604); NeurIPS 2025 poster #116510

**Authors / Institution** Sifan Wang (Yale Institution for Foundation of Data Science), Ananya Kumar Bhartari, Paris Perdikaris (UPenn MEAM), 외

**혁신** Pinn 학습의 핵심 병목인 **loss term 간 그래디언트 충돌**을 정량화하는 **gradient alignment score** 도입; **quasi-second-order** 옵티마이저(특히 SOAP)가 명시적 loss weighting 없이 충돌을 해소함을 이론·실험으로 입증. “state-of-the-art results on 10 challenging PDE benchmarks, including the first successful application of Pinns to turbulent flows at Reynolds numbers up to 10,000”.

**Why notable** Pinn의 “stiff PDE에서 안 된다”는 가장 강한 비판을 정면 격파.

**Link** <https://arxiv.org/abs/2502.00604>

### 2.4.2 PIKAN vs PINN Unified Benchmark

**arXiv** [arXiv:2602.15068](https://arxiv.org/abs/2602.15068) (Feb 14, 2026)

**Authors / Institution** Salvador K. Dzimah (MIT), Sonia Rubio Herranz, Fernando Carlos López Hernández, Antonio López Montes (Universidad Complutense de Madrid)

**혁신** 동일 파라미터 예산·동일 물리 손실 하에 MLP-Pinn vs **Physics-Informed Kolmogorov-Arnold Networks(PIKANs)** 통제 비교. PIKAN이 더 적은 **iteration**으로 더 높은 정확도 + 더 **정확한 gradient** 추정 — 특히 oscillatory-sharp-gradient 문제에서 차이가 크다.

**Link** <https://arxiv.org/abs/2602.15068>

### 2.4.3 Lang-Pinn — LLM Multi-Agent PINN Builder (ICLR 2026 submission)

**arXiv** [arXiv:2510.05158](https://arxiv.org/abs/2510.05158) (Oct 3, 2025)

혁신 자연어 task description → PDE 추출(PDE Agent) → 아키텍처 선택(PINN Agent) → 코드 생성(Code Agent) → 디버깅(Feedback Agent)의 4-에이전트 LLM 파이프라인.

Link <https://arxiv.org/abs/2510.05158>

#### 2.4.4 AB-PINNs — Adaptive-Basis Residual-Driven Domain Decomposition

arXiv [arXiv:2510.08924](https://arxiv.org/abs/2510.08924) (Oct 10, 2025)

**Authors / Institution** Jonah Botvinick-Greenhouse (Cornell), Wael H. Ali (MERL), Mouhacine Benosman, Saviz Mowlavi (MERL, co-corresponding)

혁신 도메인 분해를 고정 분할(XPINN 등)이 아니라 학습 중 residual loss가 높은 곳에 동적으로 새 서브도메인을 spawning.

Link <https://arxiv.org/abs/2510.08924>

#### 2.4.5 Architectural Physics Embedding for Wave Field Reconstruction

arXiv [arXiv:2603.02231](https://arxiv.org/abs/2603.02231) (Feb 13, 2026)

**Authors** Huiwen Zhang, Feng Ye, Chu Ma

혁신 Helmholtz 방정식의 fundamental solution(점원 응답)을 네트워크 아키텍처에 하드 임베딩.

Link <https://arxiv.org/abs/2603.02231>

#### 2.4.6 P-PINN — Selective Pruning for Noisy Inverse Problems

arXiv [arXiv:2602.19967](https://arxiv.org/abs/2602.19967) (Feb 2026)

**Authors / Institution** Zhejiang University + University of Pennsylvania (Y. Chen) + Texas Tech

혁신 학습 후 activation-level post-hoc pruning(일종의 “machine unlearning”)으로 noisy data로부터 학습한 spurious feature 제거. PDE inverse-problem 벤치마크에서 베이스라인 Pinn 대비 최대 96.6% 상대 오차 감소.

#### 2.4.7 Diffusion Models with Physics-Guided Inference

arXiv [arXiv:2604.01242](https://arxiv.org/abs/2604.01242) (Apr 2026)

혁신 확산 모델(diffusion model)의 reverse process에 Pinn-style PDE residual을 guidance로 주입.

#### 2.4.8 Rethinking Input Domains via Geometric Compactification Mappings

arXiv [arXiv:2602.16193](https://arxiv.org/abs/2602.16193) (Feb 2026)

혁신 Pinn에서 무한·반무한 도메인을 다루는 표준 해법(절단/패딩)을 **conformal/diffeomorphic compactification mapping**으로 대체.

#### 2.4.9 HyResPINNs — Hybrid Residual PINN with Adaptive Function Spaces

arXiv [arXiv:2410.03573](https://arxiv.org/abs/2410.03573) (v2 late 2025)

혁신 고정 basis(Fourier/RBF) 또는 고정 도메인 분해가 아니라 학습 중 **expressive function space** 자체를 동적으로 확장.

#### 2.4.10 Learnable Loss Balancing + Transfer Learning PINN (ICLR 2026 submission)

OpenReview [1dNbK58bB9](https://openreview.net/forum?id=1dNbK58bB9) (Sept 20, 2025 modified Oct 8, 2025)

혁신 손실 항의 가중치를 **learnable blending neuron**으로 처리 + transfer learning. CFD 87 데이터 포인트에서 <8% 오차 달성.

#### 2.4.11 Training Deep Physics-Informed Kolmogorov-Arnold Networks

arXiv [arXiv:2510.23501](https://arxiv.org/abs/2510.23501) (Oct 2025)

혁신 깊은 PIKAN의 학습 실패 원인 분석 + **basis-agnostic Glorot-like 초기화** 스킴 제안.

#### 2.4.12 Saddle-Point Reformulation for PINN Stability

arXiv [arXiv:2507.16008](https://arxiv.org/abs/2507.16008) (2025)

혁신 손실 가중치  $\pi$ 를 외부 하이퍼파라미터가 아닌 **min-max saddle-point 게임**으로 재정의.

### 2.5 관찰된 트렌드

1. 임상에서의 “환자별 **retraining 제거**”: GNN / operator learning / explicit representation 3가지 경로로 Pinn의 가장 큰 임상 장벽 제거.
2. UQ가 **default**: PUNCH의 variational inference, \$PINN의 Bayesian domain decomposition, Causal Inverse Pinn 등.
3. 2차 최적화의 **부활**: SOAP/Newton류가 Pinn의 stiff/turbulent 한계를 깬.
4. KAN의 정착: PIKAN/cPIKAN이 MLP 백본 대비 우위로 확립.
5. LLM이 Pinn 개발 루프에 진입: Lang-Pinn 등으로 PDE 명세→코드 자동화.
6. **Architectural physics embedding 부상**.
7. **혈류 특이 흐름**: (a) 4D Flow MRI super-resolution의 GS 전환, (b) raw signal 학습(SinoFlow), (c) variational inference 기반 inverse 솔루션 표준화(PUNCH).

## 2.6 Recommendations (CTO 관점)

### Stage 1 — 0-3개월: 내부 기준선 정립

- Pinn 학습 인프라 표준화: SOAP 옵티마이저 + AB-PINN 동적 도메인 분해 + PIKAN(Chebyshev) 백본의 3축
- UQ를 기본값으로: variational inference 또는 BPINN으로 신뢰구간 동반 출력
- 기준 데이터셋 확보: BenchAnXplore — 105 두개내동맥류 CFD 벤치마크

### Stage 2 — 3-9개월: 제품 라인 선택

- 하나의 임상 use-case 선택: PUNCH (관상동맥 CFR) 또는 Lannelongue GNN (두개내동맥류)
- PINGS-X 패턴 채용: 자체 4D Flow MRI 파이프라인이 있다면 axes-aligned Gaussian Splatting으로 전환

### Stage 3 — 9-18개월: 차세대 R&D

- Foundation Pinn 검토
- LLM 에이전트화: Lang-Pinn 패턴
- Architectural physics embedding 실험

### 의사결정 임계값 (벤치마크)

- 환자별 추론 시간 >5분이면 GS/operator learning으로 즉시 전환
- WSS/FFR 추정 calibration ECE > 0.1이면 UQ 미탑재로 간주, 임상 사용 보류
- Stiff/turbulent PDE에서 Pinn이 발산하면 SOAP 옵티마이저 우선 시도

## 2.7 Caveats

### 일본 리서치의 caveat 섹션

1. arXiv preprint 비중이 큼: 2026년 1-5월의 많은 논문이 peer review 통과 전.
2. 임상 검증 한계: PUNCH는 환자 20명; Lannelongue BenchAnXplore는 105 geometry. 다기관 prospective trial은 모두 미진행.
3. Pinn의 근본 한계가 완전히 해결된 것은 아님: turbulent flow  $Re > 10^4$ 는 여전히 도전 영역.
4. AAAI 2026·ICLR 2026 일부 논문은 acceptance 미확정.
5. DCP-INN의 소속 불확실.
6. PUNCH는 산업체 주도.
7. 혈류 Pinn 리뷰 논문 (ScienceDirect S0952197625028659)이 본 보고서에 부분적 기반.
8. “200× 속도” 등 수치는 저자 보고 기준이며 production end-to-end는 다를 수 있음.

# Chapter 3

## Lang-PINN 첫 번째 해설

### 3.1 왜 이 논문이 중요한가 — 해결하려는 문제

기존 LLM 기반 Pinn 자동화 도구들(CodePDE, PINNsAgent 등)의 공통 가정은 “PDE는 이미 formal하게 주어진다”였습니다. 즉 사용자가  $\partial_t u = \kappa \partial_{xx} u$  같은 형식으로 방정식을 적어줘야 하고, LLM은 그 다음 단계(아키텍처 선택, 코드 생성)만 도와줍니다.

저자들은 이 가정이 깨질 때 무슨 일이 벌어지는지 통제 실험으로 증명합니다. PINNacle 벤치마크에서 8개 PDE를 뽑아 동일한 PDE를 4단계의 언어 표현 변이(level 1 = 논문체, level 4 = 흩어진 lab notebook 노트)로 다시 쓴 1,600개 description-PDE pair 데이터셋 Task2PDE를 만들고 Llama2, Vicuna, DeepSeek-V3, Qwen으로 테스트하면, 언어가 자연스러워질수록 PDE 추출의 symbolic equivalence가 가파르게 떨어집니다.

이게 문제인 이유: 과학자가 “열원이 도메인 중심에서 경계로 이동했다”고 자연어로 말하면 source term + BC + spatial dependence 3개가 동시에 바뀌어야 하는데, 이 manual translation이 전체 Pinn 워크플로의 가장 큰 병목입니다. Lang-Pinn은 이 upstream 단계까지 자동화하는 첫 end-to-end 시스템입니다.

### 3.2 4-Agent 아키텍처

전체 파이프라인은 sequential하게 흐르지만 Feedback Agent가 closed loop으로 upstream을 수정합니다. RACI 관점에서 보면 PDE Agent가 R(formulation), PINN Agent가 R(architecture), Code Agent가 R(implementation), Feedback Agent가 A(accountable for quality) + C/I(consulted/informed across all)의 역할.

#### 3.2.1 Agent 1: PDE Agent — Linguistic Variability를 격파하는 방법

핵심은 K개의 CoT trajectory를 샘플링한 후 consensus voting으로 정답을 정합니다.

1. Reasoning-Selection 파이프라인: 자연어 description  $d$ 를 받으면, K개의 chain-of-thought를 독립적으로 sampling. 각 trajectory가 노이즈를 정리한 normalized description  $\hat{d}_k$ 를 만들고 canonical PDE candidate  $E_k$ 를 생성.

2. **Template validation:** operator well-formedness, residual form, admissible BC/IC 조건으로 invalid candidate 필터링.

3. **Symbolic Equivalence:** 각 후보 PDE를 Sympy로 AST로 파싱. 두 후보의 tree-matching score:

$$\text{sym}(E_i, E_j) = \frac{|M(T(E_i), T(E_j))|}{\max(|T(E_i)|, |T(E_j)|)}$$

4. **Semantic Consistency:** 각 후보를 normalized summary  $g(E)$ 로 paraphrase한 후 embedding cosine similarity 또는 LLM entailment score로 비교.

5. **Composite Score:**  $S(E_i, E_j) = \alpha \cdot \text{sym} + (1 - \alpha) \cdot \text{sem}$ . 0.80 threshold가 calibrated 값.

### 3.2.2 Agent 2: PINN Agent — Training-Free Architecture Selection

저자들이 사전 실험으로 보여준 핵심 관찰: 어떤 아키텍처도 모든 PDE에서 우수하지 않다.

방식은 history reuse + knowledge-guided matching의 2단계:

#### 1) PDE Feature Vector $\phi(E)$ (3차원)

- **Periodicity**  $f_{per}(E) = |P(E)|/d$
- **Geometry complexity**  $f_{geo}(E) = \text{clip}(0.6c_\Omega + 0.4c_{disc})$
- **Multi-scale demand** logistic 매핑

2) **Architecture Capability Vector  $\psi(A)$  (3차원):** Bradley-Terry model로 pairwise win-loss를 통한 ability parameter 추정. 두 측정의 분산으로 가중치 결정:

$$\omega_k = \sigma_{Abs}^2 / (\sigma_{Abs}^2 + \sigma_{BT}^2)$$

3) **Matching:**  $S(A, E) = (W\phi(E))^T \psi(A) / (\|W\phi(E)\| \|\psi(A)\|)$  weighted cosine similarity.

### 3.2.3 Agent 3: Code Agent — Modular Generation + PDE Loss Verification

핵심 통찰: **monolithic code generation**은 한 번 실패하면 전체 재생성이 필요해서 **fragile**. 6개 PDE 통제 실험에서 modular generation이 monolithic 대비 success rate 2배 이상.

가장 영리한 부분은 **PDE loss verification**: 생성된 loss 코드를 다시 symbolic PDE  $\hat{E}$ 로 parse-back해서 PDE Agent가 준  $E$ 와 equivalence를 체크.

### 3.2.4 Agent 4: Feedback Agent — Closed Loop의 핵심

두 가지 모드로 작동:

- **Mode 1: Error localization**

- **Mode 2: Multi-dimensional quality evaluation** – effectiveness, efficiency, robustness 평가

가중 합:  $S(C) = \sum_i w_i \hat{m}_i(C)$

**Iterative refinement:**  $S(C^{(t)}) > S(C^{(t-1)})$ 이면 accept, 아니면 rollback.

### 3.3 실험 결과 정리

PINNacle 14개 PDE(1D~ND), DeepSeek-V3 backbone, 10 runs × 최대 30 refinement cycle.

- **MSE 감소:** KS·Poisson-MA·Heat-ND에서 **3 orders of magnitude 이상 감소**
- **Success rate:** 1D/2D에서 80% 이상, 3D 75%, ND 73-76%
- **Iterations to first runnable:** 평균 8 iterations (worst baseline 31 대비 **74% 감소**)
- **Wall-clock time:** 평균 199.7초 (RandomAgent 413.5초의 **52% 감소**)

**Ablation별 기여:**

- Modular code generation → success rate +22%
- Metric-guided feedback → MSE  $2.5\times \downarrow$
- PDE Agent → semantic consistency +18%

### 3.4 한계와 비판적 평가

1. **혈류 분야 직접 적용은 미검증:** PINNacle 표준 PDE만 검증
2. **Architecture pool이 제한적:** MLP, CNN, GNN, Transformer 4가지만
3. **Pinn training의 진짜 어려움(stiff PDE, turbulent flow)에는 침묵**
4. **PDE Loss verification의 round-trip이 LLM의 inverse parsing 능력에 의존**
5. ICLR 2026 under review 상태

### 3.5 CTO 관점에서의 인사이트

살펴볼 만한 multi-agent system 설계 패턴:

- **Sequential pipeline + escalating feedback**
- **Round-trip verification**
- **Consensus voting을 다중 기준으로**
- **Bootstrap variance 기반 fusion 가중치**

- **Modular generation의 success rate 효과**

## Chapter 4

# Lang-PINN 심화 해설 — Architecture, Math, Critique

### 4.1 논문의 출발점 — 세 가지 통제 실험 결과

저자들은 method를 제안하기 전 Section 3에서 Pinn 파이프라인의 세 모듈을 따로 측정합니다. 이 사전 실험이 4-agent 구조를 정당화하는 근거가 됩니다.

#### 4.1.1 실험 ① — Linguistic Variability of Task Formulation

8개 PINNacle PDE를 뽑아 4단계 언어 복잡도로 재표현한 1,600개 description-PDE pair (Task2PDE)에서 측정. 결과: **Level 1 → Level 4로 갈수록 모든 모델에서 정확도가 단조 감소**. DeepSeek-V3조차 Level 4에서 약 60% 수준.

#### 4.1.2 실험 ② — Variability of Architecture Performance Across PDEs

MLP, CNN, GNN, Transformer 4가지를 Shallow Water·Convection·Poisson·Heat에 적용한 MSE 비교 결과:

- **Convection·Heat**: CNN과 Transformer가 우세
- **Poisson**: MLP와 GNN이 lowest error
- **Shallow Water**: 차이 미미

#### 4.1.3 실험 ③ — Complexity of Code Generation

6개 PDE에서 monolithic vs modular code generation 비교. **modular이 monolithic 대비 2배 이상 success rate**.

## 4.2 4-Agent 시스템의 수학적 정밀화

### 4.2.1 PDE Agent — 두 개의 직교적 유사도 척도

Symbolic Equivalence (AST 매칭)

$$\text{sym}(E_i, E_j) = \frac{|M(T(E_i), T(E_j))|}{\max(|T(E_i)|, |T(E_j)|)}$$

Semantic Consistency (embedding-based)

$$\text{sem}(E_i, E_j) = \sigma(g(E_i), g(E_j))$$

Composite Score

$$S(E_i, E_j) = \alpha \cdot \text{sym}(E_i, E_j) + (1 - \alpha) \cdot \text{sem}(E_i, E_j)$$

**calibrated threshold = 0.80** — 200 equivalent pair와 200 non-equivalent pair로 sweep해 결정.

**Appendix 3의 검증 실험:** 같은 PDE를 perfect(C1) → notation variation(C2) → coefficient error(C3) → missing/incorrect terms(C4) → structural hallucination(C5)로 단계적 perturbation 했을 때, semantic score와 PINN training의 최종  $-\log_{10}$  MSE의 **Pearson correlation**이 0.88.

### 4.2.2 PINN Agent — Bradley-Terry로 architecture 능력 측정

**PDE Feature Vector**  $\phi(E) = [f_{per}, f_{geo}, f_{ms}]^\top$

- **Periodicity**  $f_{per}(E) = |P(E)|/d$
- **Geometry complexity**  $f_{geo}(E) = \text{clip}(\lambda_\Omega c_\Omega + \lambda_{disc} c_{disc}, 0, 1) - \lambda_\Omega = 0.6, \lambda_{disc} = 0.4$
- **Multi-scale demand:**

$$\tilde{f}_{ms}(E) = 0.8 \cdot \mathbb{1}_{m \geq 3} + 0.8 \cdot \mathbb{1}_{NL=1} + 0.4 \cdot \log(1 + Re + Pe) + 1.0 \cdot \mathbb{1}_{FR=1}$$

$$f_{ms}(E) = \sigma(\tilde{f}_{ms}(E))$$

**Architecture Capability Vector**  $\psi(A) = [a_{per}, a_{geo}, a_{ms}]^\top$

*Absolute Capability:*

$$\bar{y}_k(A) = \frac{1}{|\mathcal{E}_k|} \sum_{E \in \mathcal{E}_k} y(A, E), \quad a_k^{Abs}(A) = 1 - \tilde{y}_k(A)$$

*Relative Capability (Bradley-Terry):*

$$p_{ij} = P(A_i \succ A_j) = \frac{\exp(\theta_k(A_i))}{\exp(\theta_k(A_i)) + \exp(\theta_k(A_j))}$$

likelihood

$$\mathcal{L}(\theta_k) = \prod_{i < j} p_{ij}^{n_{ij}} (1 - p_{ij})^{n_{ji}}$$

를 maximize.

*Fusion:*

$$a_k(A) = \omega_k \cdot a_k^{BT}(A) + (1 - \omega_k) \cdot a_k^{Abs}(A)$$

$$\omega_k = \frac{\sigma_{k,Abs}^2}{\sigma_{k,Abs}^2 + \sigma_{k,BT}^2}$$

즉 더 stable한(variance가 작은) estimator에 더 큰 가중. RLHF의 preference learning 인사이트를 architecture 추천에 가져온 영리한 cross-pollination.

*Matching:* weighted cosine

$$S(A, E) = \frac{(W\phi(E))^T \psi(A)}{\|W\phi(E)\|_2 \cdot \|\psi(A)\|_2}$$

### 4.2.3 Code Agent — Round-Trip Symbolic Verification

가장 영리한 부분: 생성된 loss 코드를 다시 symbolic AST로 parse-back해서 PDE Agent가 준  $E$ 와 equivalence를 체크하는 **round-trip 검증**.

6개 모듈은 표준화된 input-output interface로 연결:

- model definition → `nn.Module` with `forward(x, t)` signature
- PDE loss → scalar tensor given model and collocation points
- data preprocessing → DataLoader-compatible objects
- training loop → consumes loss + data + model
- validation → consumes model + analytic solution if available
- main → orchestrates above

### 4.2.4 Feedback Agent — 4-Dimensional Quality Score

(i) Convergence Efficiency:

$$T_{conv} = \min\{t \mid L_t \leq \tau\}, \quad m_{conv} = 1/T_{conv}$$

$$\hat{m}_{conv} = (T_{max} - T_{conv}) / (T_{max} - T_{min})$$

(ii) **Predictive Accuracy:**  $m_{acc} = -\text{MSE}(N_\theta, E)$

(iii) **Model Complexity:**  $m_{comp} = \#\text{Params}(N_\theta) / \max(\#\text{Params})$

(iv) **Robustness:**

$$m_{smooth} = 1 - \frac{\text{Std}(\Delta L_t)}{\text{Mean}(L_t)}$$

$$m_{grad} = \mathbb{1}\{\epsilon \leq \|\nabla_\theta L\|/d \leq \kappa\}$$

$$m_{rob} = \alpha \cdot m_{smooth} + (1 - \alpha) \cdot m_{grad}$$

**Aggregate Score:**

$$S(C) = w_1 \hat{m}_{conv} + w_2 \hat{m}_{acc} + w_3 \hat{m}_{comp} + w_4 \hat{m}_{rob}$$

## 4.3 Task2PDE 벤치마크 — 구체 예시

### 4.3.1 Level 1 (clean, paper-style)

“We consider one-dimensional heat diffusion in a rod of length  $L = 1$  with constant thermal conductivity  $\kappa = 0.01$ . The temperature  $u(x, t)$  satisfies  $\partial_t u = \kappa \partial_{xx} u$ ,  $x \in [0, 1]$ ,  $t > 0$ . BC:  $u(0, t) = 0$ ,  $u(1, t) = 1$ ; IC:  $u(x, 0) = \sin(\pi x)$ .”

→ direct translation, error 거의 없음.

### 4.3.2 Level 2 (irrelevant but realistic side information)

“...The lab was quite cold in the morning and the left end of the setup felt a bit colder when I touched it, but this is just due to the room air and is not part of the mathematical model. In the simulation, we still impose...”

→ 환경 묘사가 가짜 boundary condition으로 오인될 risk.

### 4.3.3 Level 3 (ambiguous shorthand)

“...At the left end, the temperature reading tends to drift over time because the sensor is not very stable, but in the actual experiment the boundary itself is kept at a fixed  $0^\circ\text{C}$  throughout the run...”

→ “drift over time”이 measurement noise인지 time-dependent BC인지 모호.

### 4.3.4 Level 4 (disorganized lab notebook)

“For this batch of runs we use the same basic heat diffusion setup as before on a rod from  $x = 0$  to  $x = 1$ . Initially we tried several values for the thermal conductivity, like  $\kappa = 0.005$  and  $\kappa = 0.02$ , but in the final configuration we fixed it at  $\kappa = 0.01$ . The initial temperature profile is the sine-shaped one from our earlier tests...”

→ 정보가 흩어져 있고 preliminary value가 final value보다 먼저 등장.

## 4.4 실험 결과 — Per-PDE Breakdown

PDE	차원	강한 baseline MSE	Lang-PINN MSE	비교
Burgers	1D	1.10E-04 (PINNsAgent)	<b>6.48E-05</b>	약 40% 개선
KS	1D	1.04E+00 (PINNacle)	<b>1.62E-03</b>	3 orders
Wave-CG	2D	2.11E-02 (BayesianAgent)	<b>2.52E-03</b>	약 10×
Heat-CG	2D	8.53E-04 (PINNacle)	1.35E-03	baseline에 약간 못 미침
NS-C	2D	1.40E-05 (PINNsAgent)	4.05E-05	baseline에 못 미침
GS	2D	4.03E-03 (BayesianAgent)	<b>1.89E-03</b>	2×
Poisson-MA	2D	1.83E+00 (PINNacle)	<b>2.25E-03</b>	3 orders
Heat-ND	ND	1.18E-04 (BayesianAgent)	4.72E-04	baseline에 못 미침

**Table 4.1:** Lang-PINN 주요 PDE의 MSE 비교 (강한 baseline 대비)

**Success Rate** (1D/2D/3D/ND 평균): Lang-PINN 73-84% vs Random/Bayesian/PINNsAgent 22-37%.

**Time Overhead:** Lang-PINN 평균 8 iterations. End-to-end wall-clock: 199.7s vs RandomAgent 413.5s (52% 감소).

**Ablation 정량 결과:**

- Modular code generation → success rate +**22%**
- Metric-guided feedback → mean MSE **2.5×** 감소
- PDE Agent’s semantic-symbolic validation → +**18%**

**LLM Backbone Robustness:** DeepSeek-V3, Qwen2, LLaMA2-Chat 모두에서 Lang-PINN이 baselines(DeepSeek-V3 사용)보다 우수.

## 4.5 Sibling Systems와의 비교

- **CodePDE** (Li et al., [arXiv:2505.08783](https://arxiv.org/abs/2505.08783), 2025) — 단일 agent + self-debugging. PDE는 formal하게 주어진다고 가정.
- **PINNsAgent** (Wuwu et al., 2025) — architecture suggestion과 code generation, PDE formulation 미적용.

- **AgenticSciML** (Jiang & Karniadakis, [arXiv:2511.07262](https://arxiv.org/abs/2511.07262), 2025) — SciML 전반의 emergent discovery.
- **PDE-Agent** ([arXiv:2512.16214](https://arxiv.org/abs/2512.16214), 2025/26) — toolchain-augmented multi-agent.
- **AutoNumerics** ([arXiv:2602.17607](https://arxiv.org/abs/2602.17607), Feb 2026) — PDE-agnostic.

#### Lang-Pinn의 차별점:

1. End-to-end NL→executable Pinn의 완전한 자동화
2. PDE loss round-trip verification
3. Bradley-Terry로 architecture capability를 데이터로 학습
4. 4차원 multi-metric feedback
5. Escalating feedback

## 4.6 Critical Assessment

### 4.6.1 강점

1. 언어 변이에 **robust**한 첫 Pinn automation
2. **Modular round-trip verification**은 software engineering 정석
3. **Bradley-Terry capability fusion**의 데이터 기반 접근
4. **Multi-metric feedback**이 **error-only feedback** 대비  $2.5\times \downarrow$

### 4.6.2 약점·한계

1. Architecture pool이 빈약: MLP/CNN/GNN/Transformer 4가지만
2. Pinn의 진짜 어려움은 다루지 않음: gradient pathology, NTK ill-conditioning
3. 혈류 등 specialized domain에 직접 적용 미검증
4. PDE Loss verification의 round-trip이 LLM의 inverse parsing에 의존
5. 30 cycle은 결코 가볍지 않음
6. ICLR 2026 under review (peer review 미통과)
7. Symbolic equivalence threshold 0.80은 PINNacle 분포에서 calibrate됨

## 4.7 Multi-Agent System Design Pattern으로서의 가치

claude-heart-beat-system처럼 5 core + 17 domain agent를 RACI로 운영하는 입장에서 transferable한 multi-agent 패턴:

- **Pattern A — Round-Trip Verification at Agent Boundary:** PDE  $\rightarrow$  code  $\rightarrow$  PDE 역변환. 일반화: “agent A의 output을 agent B가 받기 전에 A의 input domain으로 되돌려 일치 확인”.
- **Pattern B — Multi-Criterion Voting with Orthogonal Metrics:** symbolic + semantic처럼 서로 다른 failure mode를 잡는 평가 척도를 가중 결합. \*-red adversarial skill pattern과 통하는 철학.
- **Pattern C — Escalating Feedback (Non-Local Correction):** Feedback Agent가 PDE Agent까지 거슬러 올라가는 directive 전송.
- **Pattern D — Variance-Weighted Estimator Fusion:** RLHF의 BT model을 architecture selection에 가져온 cross-pollination.
- **Pattern E — Modular Generation + Standardized Interface:** Monolithic vs modular의 +22% gap.
- **Pattern F — Template Instantiation over Free-Form Generation:** 자유도가 낮을수록 reproducibility/correctness가 높다는 trade-off.

# Chapter 5

## Lang-PINN의 혈류 외 응용 도메인

Lang-Pinn의 본질은 “자연어로 기술된 PDE 문제를 자동으로 Pinn 솔루션으로 변환”하는 것이므로, PDE가 핵심 모델링 언어인 모든 도메인이 후보입니다.

### 5.1 Tier 1 — 즉시 적용 가능 (Lang-Pinn의 sweet spot)

#### 5.1.1 열전달·전자기기 thermal management

가장 자연스러운 fit. PINNacle에 Heat-CG, Heat-MS, Heat-VC, Heat-ND가 이미 포함되어 검증됨.

구체 use case:

- 데이터센터 GPU 클러스터 cooling 설계 — “12개의 H100이 3행 4열로 배치, 각각 700W 발산, 측면 inlet에서 18°C 공기” 같은 자연어 기술. ConoHa VPS 인프라 운영 경험과 직결.
- 반도체 chip-level hot spot 예측 — 3D heat equation with anisotropic conductivity
- PCB thermal simulation

#### 5.1.2 음향·진동·구조 동역학

Wave equation 계열이 PINNacle에 있고(Wave-C, Wave-CG), 산업 응용 풍부.

구체 use case:

- 건축 음향 — 일본 건축 시장과 가까움
- MEMS resonator 설계
- 자동차 NVH — 토요타·혼다 등 일본 자동차 기업에 직접 가치

#### 5.1.3 화학 반응 시스템·반응-확산

PINNacle에 Gray-Scott(GS)가 있어서 검증됨.

구체 use case:

- 제약·정밀화학 반응기 설계
- 배터리 전기화학 (lithium-ion intercalation) — 일본의 배터리 산업
- 반도체 dopant diffusion

## 5.2 Tier 2 — 적용 가능하나 약간의 확장 필요

### 5.2.1 지하수·환경 유체 (porous media flow)

Darcy/Forchheimer 방정식, Richards equation.

구체 use case:

- 지하수 contamination 추적
- 지열 발전 — 일본의 지열 잠재력은 세계 3위
- 이산화탄소 지중저장 (CCS)

### 5.2.2 옵션 가격결정·금융 PDE (Black-Scholes 계열)

구체 use case:

- 이색 옵션 (exotic option) pricing
- CVA/XVA computation
- mean-field game in financial markets

한계: 금융 PDE는 잘 정의된 formal 표기를 quants가 이미 쓰므로 자연어→PDE 변환의 가치가 상대적으로 낮음.

### 5.2.3 광학·포토닉스 (전자기 wave propagation)

Maxwell 방정식, Helmholtz equation.

구체 use case:

- 포토닉스 결정 (photonic crystal) 설계
- 메타물질 (metamaterial) 시뮬레이션
- 광섬유 통신의 nonlinear pulse propagation — 일본의 NTT·KDDI 통신 인프라

## 5.3 Tier 3 — 흥미롭지만 도전적 영역

### 5.3.1 양자역학 — Schrödinger equation

Time-dependent Schrödinger Equation의 numerical solution.

왜 도전적: 복소수 wavefunction, hermitian operator 보존, probability conservation 같은 constraint를 Pinn으로 enforce하는 게 nontrivial.

### 5.3.2 우주물리·천체역학 — General Relativity

Einstein field equation의 다양한 limit. LIGO/KAGRA의 waveform modeling.

현실적 한계: Lang-Pinn의 PDE Agent가 tensor PDE를 다루도록 학습되지 않음.

### 5.3.3 인구역학·역학 (epidemiology, ecology)

SIR/SEIR model의 reaction-diffusion 확장.

구체 use case:

- 공간적 epidemic spread
- 생태계 invasive species spread — Fisher-KPP equation

### 5.3.4 재료과학 — Phase field method

Cahn-Hilliard, Allen-Cahn equation.

구체 use case:

- 금속 alloy 응고 시뮬레이션
- 배터리 음극 SEI layer formation
- 3D printing의 melt pool dynamics

## 5.4 CTO 관점 — 어디부터 시작할까

planitai.co.jp의 컨설팅 포트폴리오와 일본 산업 구조를 고려한 우선순위:

가장 빠른 ROI (3-6개월):

- 반도체·전자기기 thermal management
- 배터리 전기화학

중기 R&D (6-18개월):

- 건축 음향·자동차 NVH

- CCS·지열

#### 탐색적 베틱 (18개월+):

- 광섬유 통신의 nonlinear optics
- 금융 PDE

#### 원칙적 캐비엣

Lang-Pinn의 진짜 가치는 “PDE를 잘 모르는 domain expert(의사, 약사, 화공 엔지니어, 건축가)가 자연어로 기술하면 Pinn이 나온다”는 점입니다. 따라서 PDE 표기가 이미 표준화된 quants나 물리학자보다, PDE를 손으로 적는 게 부담스러운 도메인에서 가장 큰 가치를 발휘합니다. 의사가 환자 hemodynamics를 기술하는 게 그 대표 케이스였고, 같은 논리로 건축가의 음향 설계, 화공 엔지니어의 반응기 설계, 재료 엔지니어의 미세조직 진화 기술이 fit이 좋은 후보입니다.

반대로 이미 formal PDE specification이 산업 표준인 영역(quant finance, computational electromagnetics)에서는 Lang-Pinn의 PDE Agent가 주는 가치가 작고, CodePDE 같은 더 단순한 도구로 충분합니다.

## 부록 A

# 컴파일 노트

### A.1 빌드 방법

이 문서는 XeLaTeX + kotex 기반입니다. TeX Live 2022 이상이면 모든 패키지가 기본 포함됩니다.

**Listing A.1:** 빌드 명령어

```
1 # 한국어글꼴 (NanumMyeongjo, NanumGothic, NanumGothicCoding) 설치필요
2 # Ubuntu: sudo apt install fonts-nanum fonts-nanum-coding
3 # macOS: brew install --cask font-nanum-gothic font-nanum-myeongjo
4
5 xelatex pinn_report.tex
6 xelatex pinn_report.tex # TOC 생성을위해두번실행
```

폰트가 없을 경우 `setmainfont` 등의 호출에서 다른 폰트로 fallback할 수 있습니다 (예: Noto Serif CJK KR).

### A.2 소스 메타데이터

- Document class: report
- 인코딩: UTF-8
- 한국어 패키지: kotex
- 글꼴 엔진: fontspec (XeLaTeX 필요)
- 색상: xcolor
- 하이퍼링크: hyperref
- 박스: tcolorbox
- 코드: listings

### A.3 원본 대화 메타데이터

- Session date: 2026-05-20
- Compiler: Claude Opus 4.7 (Anthropic)
- User: Crowdy (CTO, PlanitAI)
- Primary language: Korean (English technical terms preserved)
- Source: web browse / arXiv fetch / OpenReview fetch / extended research task

### A.4 이 문서의 한계

#### 중요

- Chapter 2의 리서치 내용은 2026-05-20 기준이며, 그 이후 paper들의 출판 여부, peer review 결과 등은 반영되어 있지 않습니다.
- arXiv preprint의 비중이 높습니다. 인용 전 venue 확정과 peer review 통과 여부를 별도 확인해야 합니다.
- PUNCH의 PMLR vol. 정보, Lannelongue의 DOI 등은 fetch 시점의 메타데이터입니다.
- LLM 기반 합성 텍스트이므로 모든 수치·인용은 원본 논문에서 재검증을 권장합니다.

---

*End of document.*