中国科协年会
Annual Meeting of the China Association for Science and Technology

Real-Time Computing in a Dynamic, Open Environment: The Broad Learning Approach for Offline Intelligent

C. L. Philip Chen

Dean, School of Computer Science and Engineering,
South China University of Technology,
Guangzhou, Guangdong, China
Co-President, Guangdong AI Industry Association
Former Vice-President, Chinese Automation Association
Former Vice-President, Macau S&T Association
Former President, IEEE SMC Society

Director, Ministry of Education, Health Intelligence and Parallel Digital Human Engineering Research Center
Director, Guangdong Computational Intelligence Modelling and Perception Key Lab
Vice Director, Guangdong AI and Digital Economy Lab
Honor President, Shenzhen 人AI Industry Association

National Distinguished Expert
IEEE Life Fellow, AAAS Fellow, IAPR Fellow, CAA, CAAI, HKIE Fellow
Member of Academia Europaea (M-AE)
Member of European Academy of Science and Arts (EASA)
Foreign member of Russian Academy of Engineering (RAE)
Philip.Chen@ieee.org

C. L. Philip Chen

Member of Academia Europaea

Member of European Academy of Science and Arts（EASA)

1985 MS University of Michigan

1988 Ph.D. Purdue University

1989-2001 Wright State University, Ohio

Visiting Research Scientist, 1990-2000, US WP AFB, Wright Lab, NASA Glenn Research Center

2002 Universityy of Texas, San Antonio

2005 UTSA, Dept head and Associate Dean

2007 IEEE Fellow/AAAS Fellow/IAPR Fellow，2015

2010, Dean of Faculty of S&T, University of Macau

2012-2013, IEEE SMC Society

2014--2019 -2021 IEEE Trans on SMC: Systems/Cybernetics Editor-in-Chief (IF 13.451/11.488)

2018 Elected member of EA&EASA

2019-, Dean, School of Computer Science and Eng

2018-2026 HCR

2021, Wu WJ AI Distinguished Contribution Award, 2018, 2021年，IEEE Norbert Weiner, IEEE Joseph G. Wohl Award

- IEEE Fellow，AAAS Fellow，IAPR Fellow，中国自动化学会Fellow
- 中国自动化学会副理事长;
- 国际总主席，IEEE系统人机及智控学会(SMCS) (2012-2013);
- IEEE SMCS Fellow 评选主席（2016-2017）、审核委员（2009-2018）;
- 现任CAAI Trans. on AI执行主编，曾任IEEE Trans. Cybernetics和IEEE Trans. on SMC: Systems期刊主编及中国科学等多个期刊副编，自动化学报副主编;
- 2018和2021两次获IEEE TNNLS最佳论文奖;
- ESI前1%高被引论文58篇、3篇入选ESI前0.1%热门论文;

- SCI期刊文章1000余篇(其中，IEEE Transactions期刊文章600余篇)，学术专著1部，授权美国专利4项;
- 2018年-2024年连续6年入选科瑞唯安全球高被引科学家；5次IEEE SMC学会杰出贡献奖得主;
- 主持科技部重点研发计划、973课题，国家自然科学基金重大研究计划培育项目等10余项基金;
- 2018年获 IEEE系统人机控制论的最高学术奖--IEEE 诺伯特•维纳奖 (Norbert Wiener Award)
- 2021年度IEEE Joseph G. Wohl（约瑟夫•沃尔）终身成就奖
- 荣获2021年度吴文俊人工智能杰出贡献奖

Guangzhou: capital of Guangdong Providence

Guangzhou
China's "Southern Gateway"

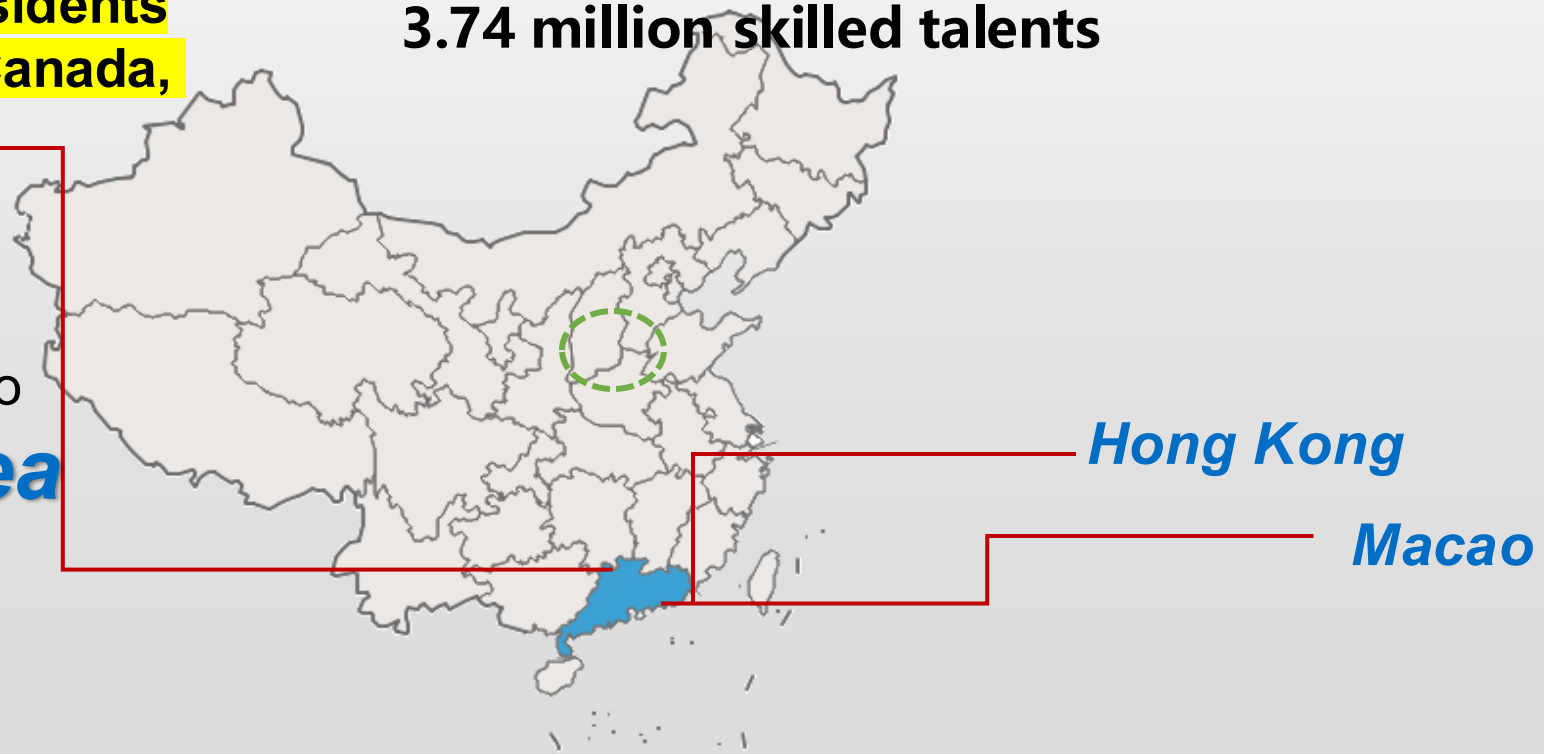
A window connecting China and the world

Population: 19.1 million residents

Guangdong Providence

- ✓ **Population: 128.1 million residents**
- ✓ **GDP: 2.05 Trillion equiv to Canada, and Russia, and Brazil**

- **The world's only major trading port that has remained prosperous for over 2,000 years**
- **One of the fastest-growing megacities in the worl**
- **13,000 high-tech enterprises, 80 new R&D institutions, 1.65 million university students, and 3.74 million skilled talents**



Guangdong, Hong Kong & Macao

Greater Bay Area

South China University of Technology
All three campuses are in Guangzhou

Wushan Campus

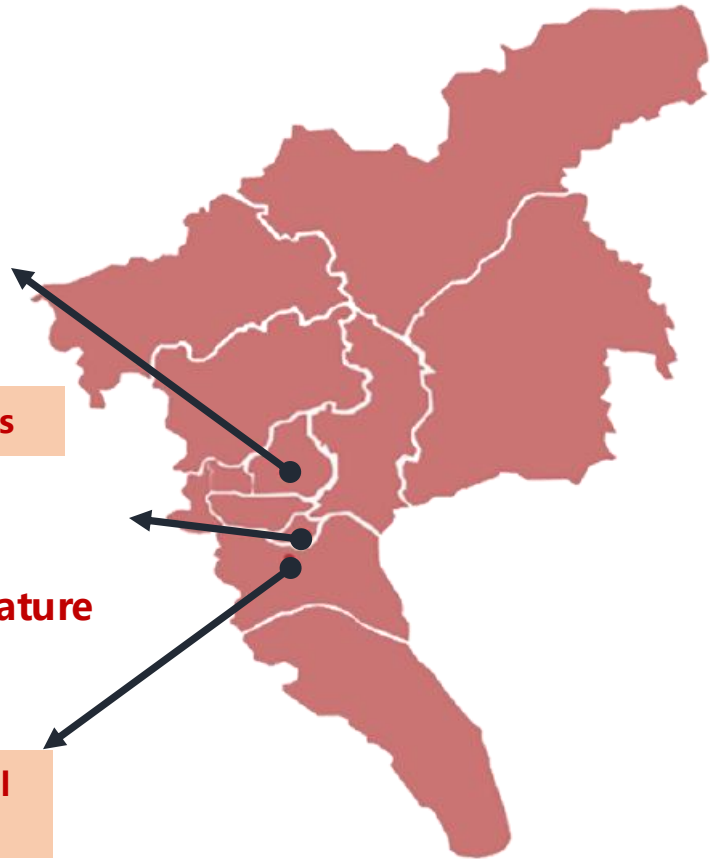
- 1.83 million㎡
- 28,000 students
- Engineering

University Town Campus

- 1.11 million㎡
- 17,000 students
- Information & Literature Arts

Guangzhou International Campus

- 1.13 million㎡
- 10,000 students
- Cross-disciplinaries



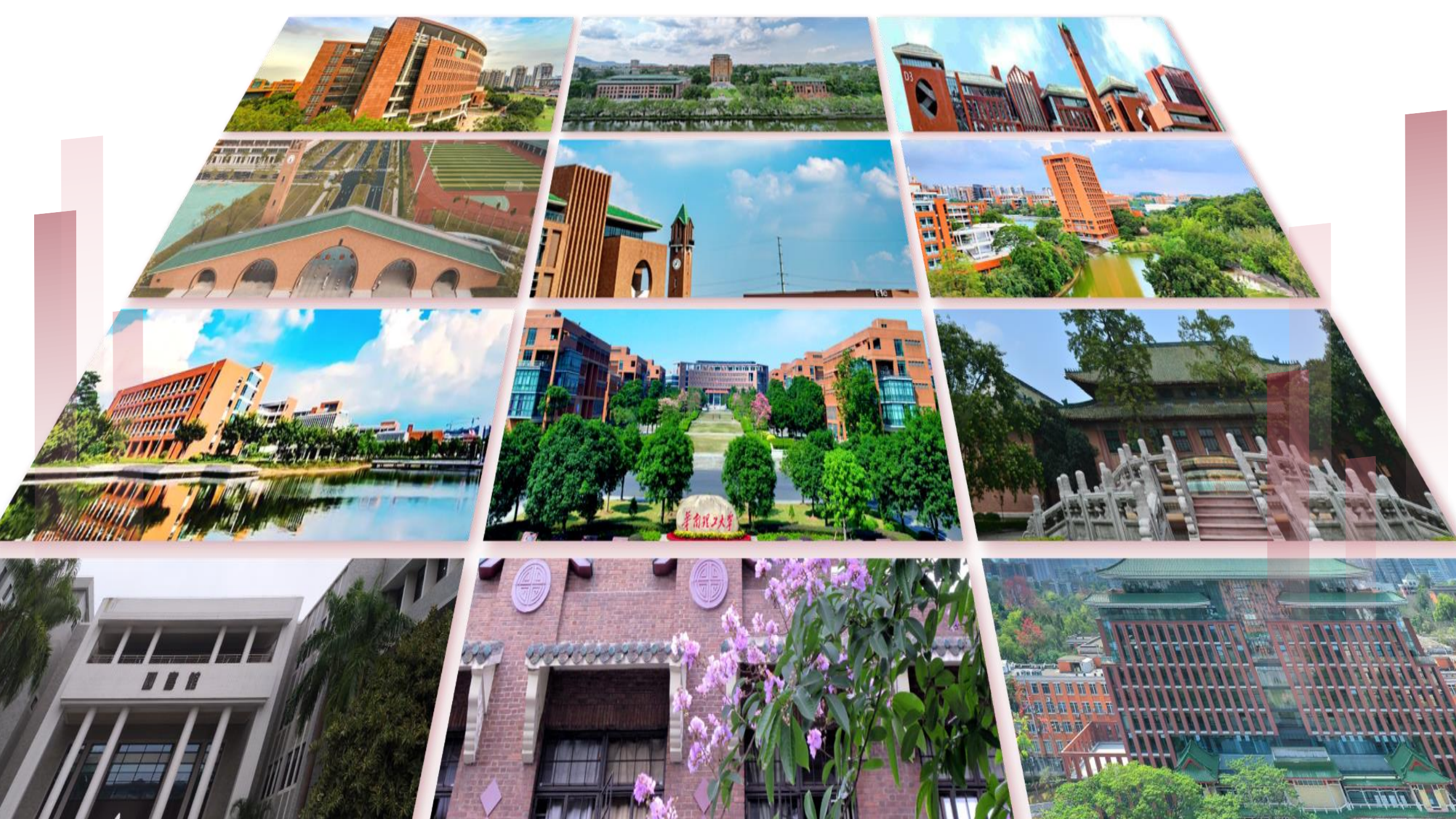
Campus size 4.47 million㎡
Building size 2.84million ㎡

Fixed assets totaled 10.433 billion yuan

4,500+ faculty and staff members

55,000+ students enrolled

37 colleges and schools


华南理工大学

South China University of Technology (SCUT)

ShanghaiRanks World University	World Top 150	2025 Global University Google Academic Impact Ranking	Mainland universities 22nd place
2024 Times Higher Education Asia University Rankings	Asian universities 45th place / Mainland universities 17th place	Nature index ranking	World Top 50

Ranking agencies	2019年	2020年	2021年	2022年	2023年	2024年
SH Rank	201-300	151-200	151-200	151-200	101-150	101-150
US NEWS	336	290	250	219	219	**187**
QS	480	462	407	406	392	**385**
THE	501-600	401-500	401-500	401-500	251-300	251-300

(There are about 45,000 universities worldwide)

Engineering ranks 22nd globally

20
Disciplines ESI top 1% globally

6
Disciplines ESI top 1‰ globally

3
Disciplines ESI top 1‰ globally

2

- Food Science and Technology
- Polymer science

2024 US News World University Subject Rankings

Top three in the world

World-class subject rankings

2 Disciplines Top 10

18 Disciplines Top 50

World University Academic Ranking

3 Disciplines Top 10

18 Disciplines Top 50

	Institutions		Cites	Cites/Pape	Top Papers
1	CHINESE ACADEMY OF SCIENCES		326716	16.99	299
2	TSINGHUA UNIVERSITY				
3	UNIVERSITY OF ELECTRONIC SCIENCE & TECHNOL(				
4	SOUTHEAST UNIVERSITY - CHINA				
5	XIDIAN UNIVERSITY				
6	UNIVERSITY OF CALIFORNIA SYSTEM				
7	NANYANG TECHNOLOGICAL UNIVERSITY				
8	UNIVERSITY OF LONDON				
9	CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQ(				
10	ZHEJIANG UNIVERSITY				
11	BEIJING UNIVERSITY OF POSTS & TELECOMMUNIC				
12	SHANGHAI JIAO TONG UNIVERSITY				
13	INDIAN INSTITUTE OF TECHNOLOGY SYSTEM (IIT S				
14	HUAZHONG UNIVERSITY OF SCIENCE & TECHNOL(				
15	NATIONAL INSTITUTE OF TECHNOLOGY (NIT SYSTE				
16	NATIONAL UNIVERSITY OF SINGAPORE				
17	HARBIN INSTITUTE OF TECHNOLOGY				
18	UNIVERSITY OF TEXAS SYSTEM				
19	WUHAN UNIVERSITY				
20	BEIHANG UNIVERSITY				
21	UNIVERSITY OF TECHNOLOGY SYDNEY				
22	EGYPTIAN KNOWLEDGE BANK (EKB)				
23	UNIVERSITY OF CHINESE ACADEMY OF SCIENCES, (				
24	ALPHABET INC.				
25	DALIAN UNIVERSITY OF TECHNOLOGY				
26	BEIJING INSTITUTE OF TECHNOLOGY				
27	CITY UNIVERSITY OF HONG KONG				
28	CENTRAL SOUTH UNIVERSITY				
29	SOUTH CHINA UNIVERSITY OF TECHNOLOGY				

层次	排名	学校名称	得分
A+	1	清华大学	73.0
A+	2	浙江大学	72.8
A+	3	哈尔滨工业大学	71.6
A+	4	北京航空航天大学	68.9
A+	5	上海交通大学	68.1
A+	6	北京大学	67.9
A+	7	华中科技大学	67.8
A+	8	南京大学	67.7
A+	8	中国科学技术大学	67.7
A+	10	北京理工大学	67.6
A+	10	西安电子科技大学	67.6
A+	12	西北工业大学	66.4
A+	13	东南大学	63.7
A+	14	同济大学	63.5
A+	15	电子科技大学	62.6
A+	16	西安交通大学	60.9
A+	17	华南理工大学	60.7
A+	18	湖南大学	60.1
A+	19	中国人民大学	60.0

关注 软科 公众号，回复 专业完整名称

30	EORGIA	USA	3696	86
31		CHINA MAINLAND	4252	86
32	TH WALES SYDNEY	AUSTRALIA	3249	86
33	M OF FLORIDA	USA	4082	85
34	CHNICAL UNIVERSITY	CHINA MAINLAND	4805	84
35	C UNIVERSITY	HONG KONG	4066	83
36		CHINA MAINLAND	4884	83
37		IRAN	4159	78
38		USA	1341	78
39	TY - CHINA	CHINA MAINLAND	4392	76
40	TECHNOLOGY OF CHINA, CAS	CHINA MAINLAND	4828	76
41	ITY	SAUDI ARABIA	2603	75
42		CHINA MAINLAND	4272	74
43	ITY	CHINA MAINLAND	4528	72
44		CHINA MAINLAND	3777	72
45		CHINA MAINLAND	3441	70
46	ON	ENGLAND	2508	69
47	WEALTH SYSTEM OF HIGHER ED	USA	3032	68
48		N/A	4861	68
49		CHINA MAINLAND	4016	67
50		CHINA MAINLAND	3865	66
51	DEFENSE TECHNOLOGY - CHINA	CHINA MAINLAND	4766	66
52	OSTS & TELECOMMUNICATIONS	CHINA MAINLAND	4226	65
53	OF SCIENCE & TECHNOLOGY	HONG KONG	2835	65
54	NFORMATION SCIENCE & TECHN	CHINA MAINLAND	3258	64
55		AUSTRALIA	2550	64
56	OLUMBIA	CANADA	1956	63

LLMs vs Small Models

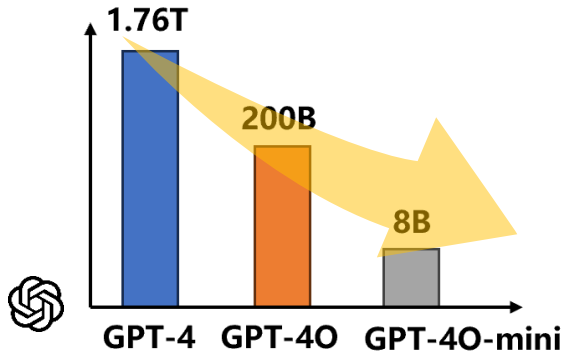
[1] Abacha, Asma Ben, Wen-wai Yim, Yujuan Fu, Zhaoyi Sun, Meliha Yetisgen, Fei Xia and Thomas Lin. “MEDEC: A Benchmark for Medical Error Detection and Correction in Clinical Notes.” ArXiv abs/2412.19260 (2024.

LLM has rapidly become a research hot spots，its application potential and industrial driving capacity have strong advantages. However, **it faces challenges** such as high computing power consumption and high deployment costs。**Lightweight large and small models are crucial for reducing computational costs and increasing computing speed.**

Lightweight large models

Compressed by large models through distillation, pruning, and other methods

- Parameter count **Less than one billion**



- Reduces computational resource consumption and increases reasoning speed
- It possesses strong reasoning abilities but relies on primitive LLM making the technology incapable of autonomous control

Pre-trained large language models

Parameter volumes ranging from tens of billions to over 100 billion

- GPT-4：1.76T
- Claude 3.5：175B
- DeepSeek V3：670B
- Qwen 1.5：110B

- Possesses strong reasoning and generation abilities, and has the ability to "emerge."

Autonomous training of small models

The number of parameters can be less than **ten million, Lowest computational costs** and **fastest inference speed**

- Task-specific training，Lack of versatility
- YOLO、MobileNet、EfficientNet、Broad learning system
- In the era of LLM, research related to small models is relatively limited

The concept of edge AI agents

Edge AI agents It is an intelligent entity deployed in edge computing environments (usually a combination of software and hardware).），posses **Perception、Decision-making、Execution** ability。It is not only the "functional unit" of edge computing but also the core of intelligent applications deployed at the edge.

- **Autonomy**：在本地完成数据处理与决策，不依赖中心云的实时反馈。
- **Intelligence**：集成机器学习模型，支持模式识别、预测、优化等任务。
- **Real-time capability**：在边缘端快速响应，满足低延迟需求。
- **Synergy**：多个边缘智能体之间可形成分布式协作网络，与云端形成“云-边-端”联动。

The structure of edge agents

章鱼就是用“边缘计算”来解决实际问题的。作为无脊椎动物中智商最高的一种动物，章鱼拥有巨量的神经元，但有高达60%分布在其八条灵活自如的触手（腕足）上，脑部却仅有40%。

边缘智能体架构

Perception layer：

传感器、摄像头、IoT设备等，负责采集环境数据。

Edge Computing Layer (Agent Core):

轻量化模型、推理引擎、缓存与本地存储。**新型边缘计算具备实时决策、自主学习、隐私保护等能力。**

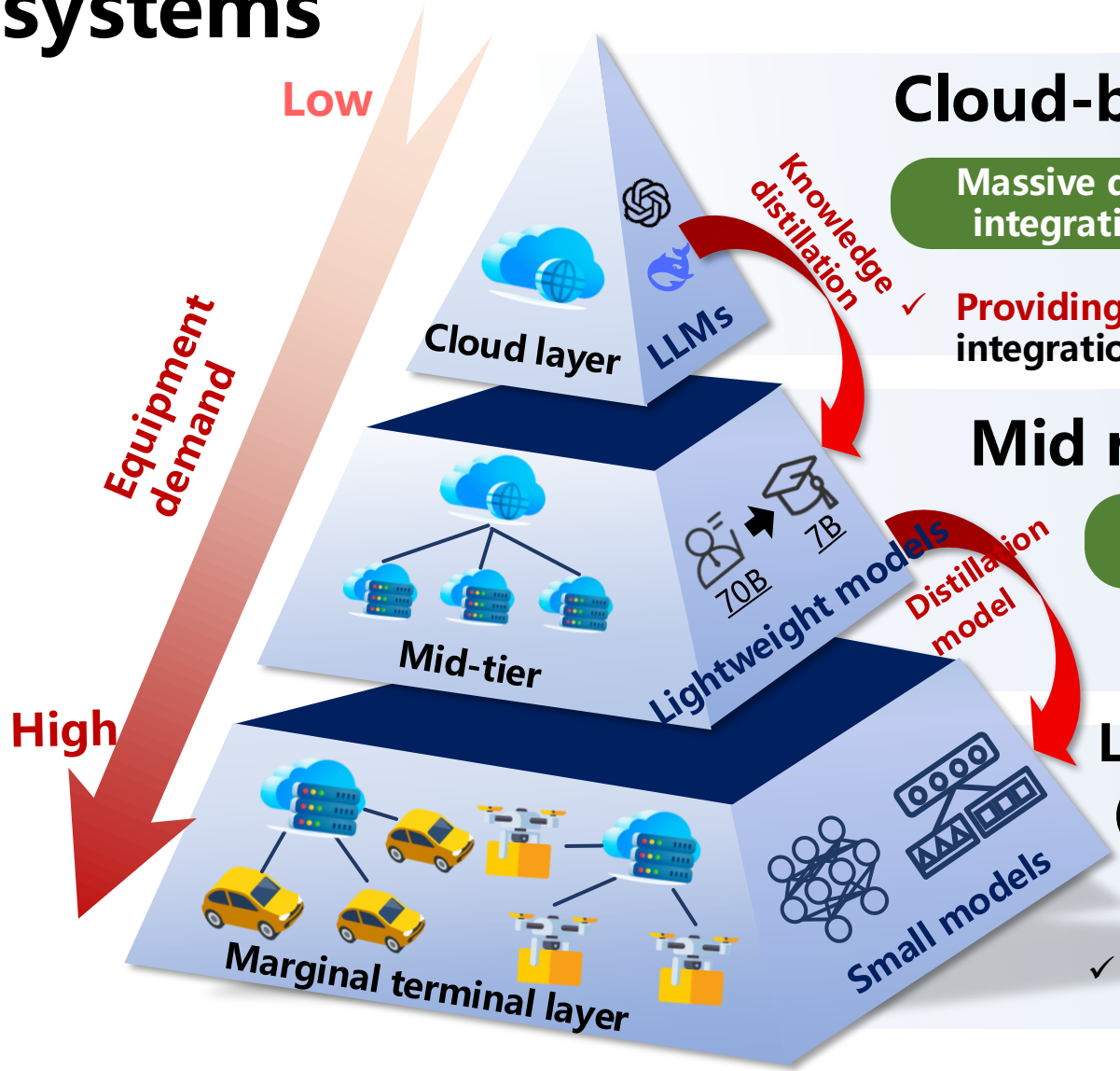
Synergy Layer:

智能体之间的横向协作（边-边），云端的纵向协作（边-云）

Execution Layer：

将智能体的决策转化为具体动作，如设备控制、告警、优化指令。

Collaborative large and small models: Cloud-edge AI systems



Cloud-based global control layer

Massive data integration → Comprehensive knowledge generation → Macro strategic planning

✓ **Providing unified data support** to achieve global knowledge integration and strategic control.

Multi-domain information integration

Mid regional collaboration layer

Regional data fine-tuning → Task adaptation → Strategy issuance

✓ Meeting **local scenario** needs to **enhance model adaptability.**

Real-time status aggregation

Lightweight layer at the edge (edge intelligence)

Lightweight deployment → Real-time → Dynamic feedback

✓ Efficient operation on the terminal enables **autonomous task execution**.

Build hierarchical, decoupled large-size models collaborating with AI systems

Industrialization application scenarios: Drone swarms

Integrating adaptive flight control and dynamic strategic planning, Enables efficient and stable cluster-based task execution.

Cloud-based knowledge

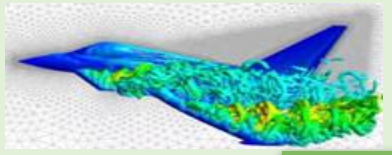
A general large model with cross-domain knowledge integration and multi-agent cluster decision-making capabilities

Knowledge distillation throughout the region

domain model

Visual domain large model

Directing and studying hardship



Road network

Communication

Flight planning, conduction, and coordination

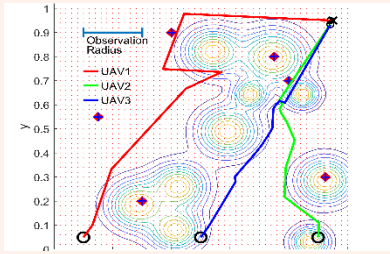
Edge-end small model

Image annotation

Flight control

Path planning

Cluster collaboration



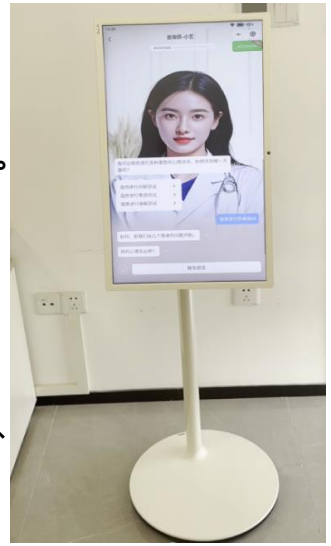
- ✓ City patrol
- ✓ Environmental monitoring
- ✓ Real-time risk avoidance

AI large models assist community school psychologist:，Digital intelligence empowers grassroots mental health (网信办立案： 4401052303629012600 17)

- **华佗之心**项目以心理大模型为技术核心，以多轮对话的形式精准洞察并响应用户的心理健康需求，衍生一系列心理健康测评系统、疗愈倾听平台、心理知识科普课堂等心理健康服务产品，全面实现心理咨询服务的数字化与多元化。

- 华银康-**华佗之心**

华银康海珠社区体检中心



- 25年实现广州多个社区试点。26年计划多省推广。
- 按照专业评测项目付费、疗愈超时长付费、个性化服务等收费方式向社区、学校、心理咨询机构收取AI心理咨询服务费。

Professional psychological knowledge base + psychological counseling scenario model

L0 General knowledge large models

L1 Mental health models | Medical knowledge large models | Multi-turn dialogue model

L2 Broad emotion recognition model | Medical knowledge recommendation model | Scale assessment model

Guidance from professional knowledge | Empathetic dialogue management | Long-term memory management

华佗之心 AI心理医生

- 覆盖全国心理咨询机构12.7万家、社区医院1.5万家、初高中学校和高校6万所。
- 提升心理咨询室服务规范达标率90%，提升居民对心理咨询服务的满意度75%
- 降低心理咨询的门槛和心理咨询成本80%，提升居民对心理咨询服务的知晓率80%，
- 解决 72% 的县域地区心理服务空白，缓解国家心理健康危机和公共医疗压力。

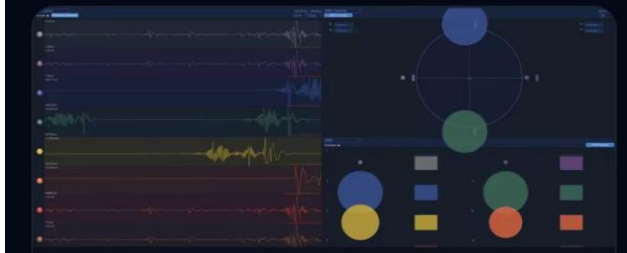
Yunova 云脑

模型、数据、硬件入口

三层技术闭环，支撑 YunovaOS 持续进化。

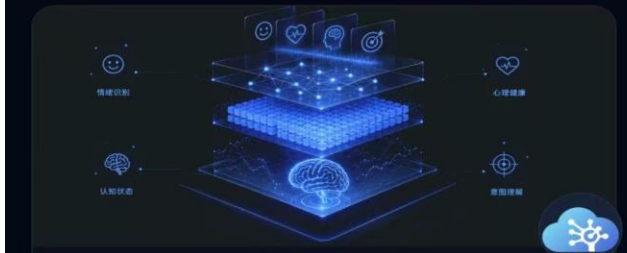
智能感知平台

脑电、肌电、眼电等感知智能穿戴。



YunovaOS系统

模型推理、云端分析。



类脑智能模型

情绪模型、精神模型、认知模型、意图模

Yunova 云脑

YunovaOS系统，连接多源信号处理

建模、推理、反馈，形成产品级数据飞轮。

情绪模型　精神模型

认知模型　意图模型

Yunova 云脑　趣乐AI

趣乐AI大模

成功通过国家大模型

国家大模型备案成

云脑智能「趣乐AI」是
全国唯一同时具备脑机接口
神经调控、AI音乐疗愈三赛
合规备案大模型。

备案实力

全国累计868款备案生成式大
专项AI音乐垂类自研模型仅5
趣乐AI为国内唯一的神经调控

技术闭环

依托双模态脑电解码技术、实现
神经反馈干预、个性化疗愈音乐
全链路闭环。

应用价值

脑健康　情绪调节

Yunova 云脑

中国科协年会
Annual Meeting of the China Association for Science and Technology

趣乐AI

国内唯一神经调控音乐大模型

成功通过国家大模型备案

打通 脑机接口 + 神经调控 + AI音乐疗愈

01 五音音乐生成

面向身心平衡与
传统五音疗愈场景，
生成个性化音乐内容

宫 商 角 微 羽

02 冥想音乐生成

用于放松减压、
专注训练与情绪舒缓

03 睡眠音乐生成

用于入睡辅助、
睡眠陪伴与夜间放松

技术优势

- 双模态脑电解码：多维解析脑电信号，精准识别身心状态
- 神经反馈干预：实时神经反馈，调节脑状态
- 个性化音乐生成：AI生成专属疗愈音乐，匹配个体需求
- 软硬一体合规布局：软硬一体系统方案，实现合规落地

未来将持续拓展更多音乐疗愈场景

图 1 平陆运河途径地示意图

图 2 平陆运河抬填造地与耕地复垦

项目名称

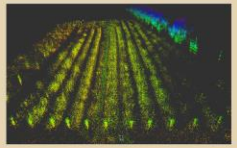
面向丘陵山区的四足农业智能巡检机器人

主要研究内容

一、多模态融合的丘陵山区混合地图构建与自主导航方法

①基于Cartographer框架的三维点云地图构建

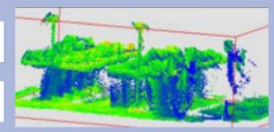
②基于点云地图的路径规划与精准导航技术



二、基于三维点云的丘陵山区作物表型无损提取模型

①基于激光雷达三维点云的作物株高测量方法

②基于视觉AI的作物诊断方法



三、动态扰动下基于手势的遥操作鲁棒跟踪控制

①基于Leap Motion的手势遥操作控制

②动态扰动下四足机器人的鲁棒跟踪控制

现阶段成效

系统集成

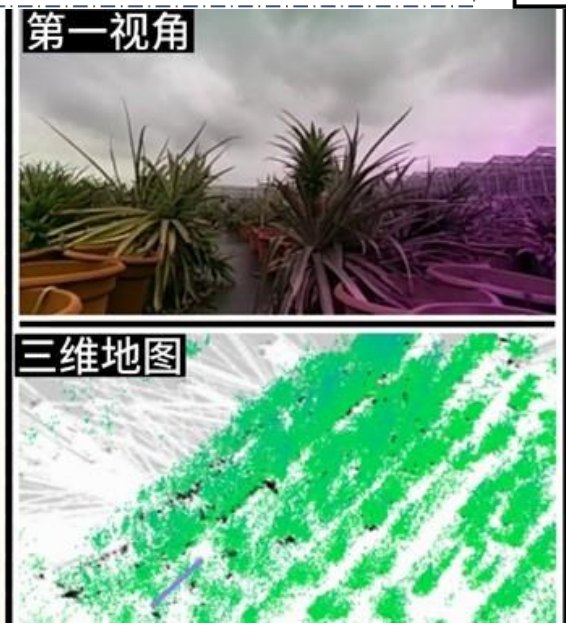
作业机器人

巡检软件平台

爬坡角度≥30°
爬梯高度≥20cm

不间断运行4h以上
运动负载≥20kg

诊断准确率≥90%
建图定位误差≤6cm



Hardware Systems

核心产品

AI servers

AI中心侧推理、模型训练

AI workstation

边缘侧AI场景高性能设备

AI accelerator card

搭载华为昇腾AI芯片平台

云-边-端全场景概念架构

云端业务平台

管理服务 | 应用服务 | 数据路由服务

边侧应用

- 智能分析
- 数据管理
- 设备运维

资源管控

- 视频管理：处理视频流采集编解码及分析实时推理
- 算力管理：硬件算法资源调度
- 文件服务：日志、管理模型文件等存储与分发
- 流量管控：优化边缘侧带宽分配

算法下发 / 数据回传

边缘计算设备

智能工作站

AI加速卡与模组

AI边缘服务器

推训一体机

端侧设备

Real-time computation in Edge Embodied Intelligence via:

Broad Learning Systems

Wu WenJun AI Outstanding Contribution Award, China

Engineering and Scientific Applications

Optimization, Regression, and Modeling

Least Square Regression

$$arg\ min_{\boldsymbol{W}}: \|\boldsymbol{Y} - \boldsymbol{AW}\|_2^2 + \lambda\|\boldsymbol{W}\|_2^2$$

$$\boldsymbol{W} = \left(\boldsymbol{A^T A} + \boldsymbol{\lambda I}\right)^{-1}\boldsymbol{A^T Y}$$

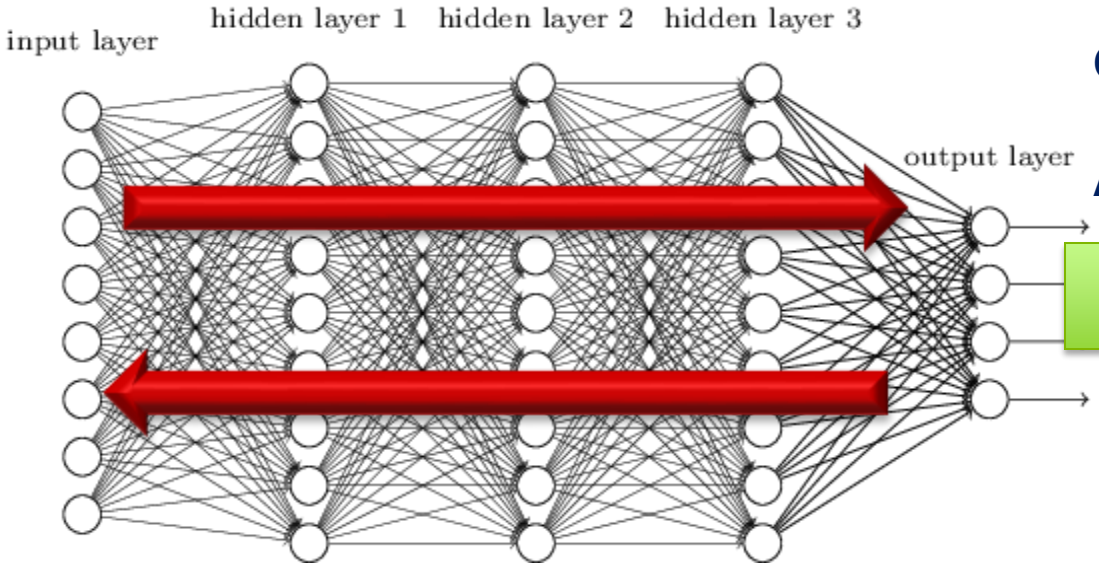
Least Square Regression

$$arg\ min_{\boldsymbol{W}}: \|\boldsymbol{Y} - \boldsymbol{AW}\|_2^2 + \lambda\|\boldsymbol{W}\|_2^2$$

$$\boldsymbol{W} = \left(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A} + \boldsymbol{\lambda I}\right)^{-1}\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{Y}$$

Almost in real systems, there exist some levels of non-linearity and uncertainty

Deep Neural Networks approach



Convert to "Broad" Architecture

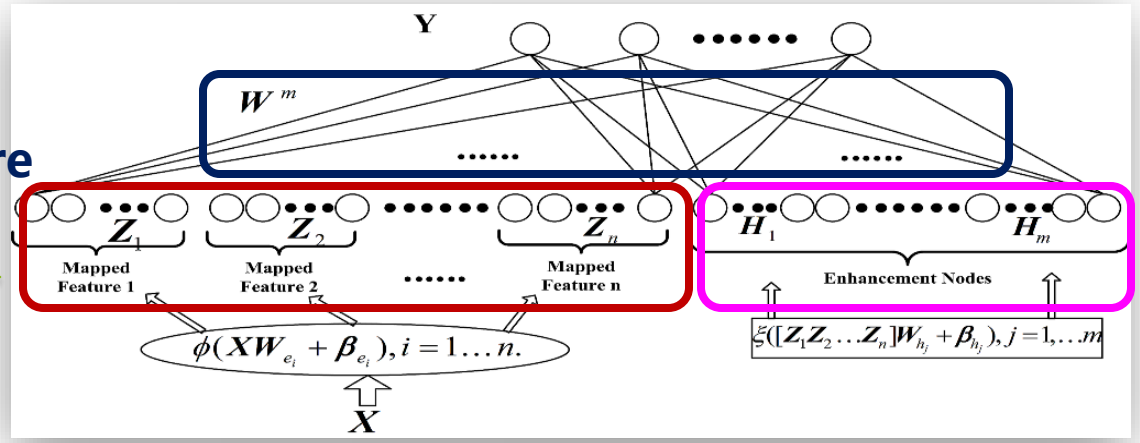
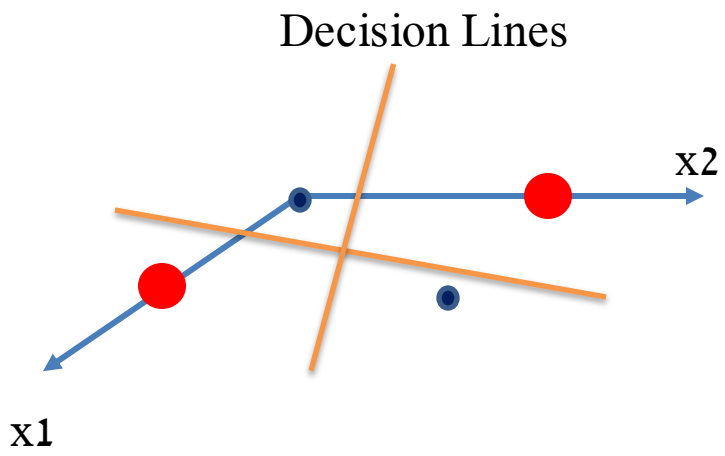


Table 3.4 - Two-Bit Parity in Two and Three Dimensional Feature Space

x_1	x_2	2-bit Parity
0	0	0
0	1	1
1	0	1
1	1	0

Figure 3.15 - Linear Separation of Two Bit Parity Data in Three Dimensions

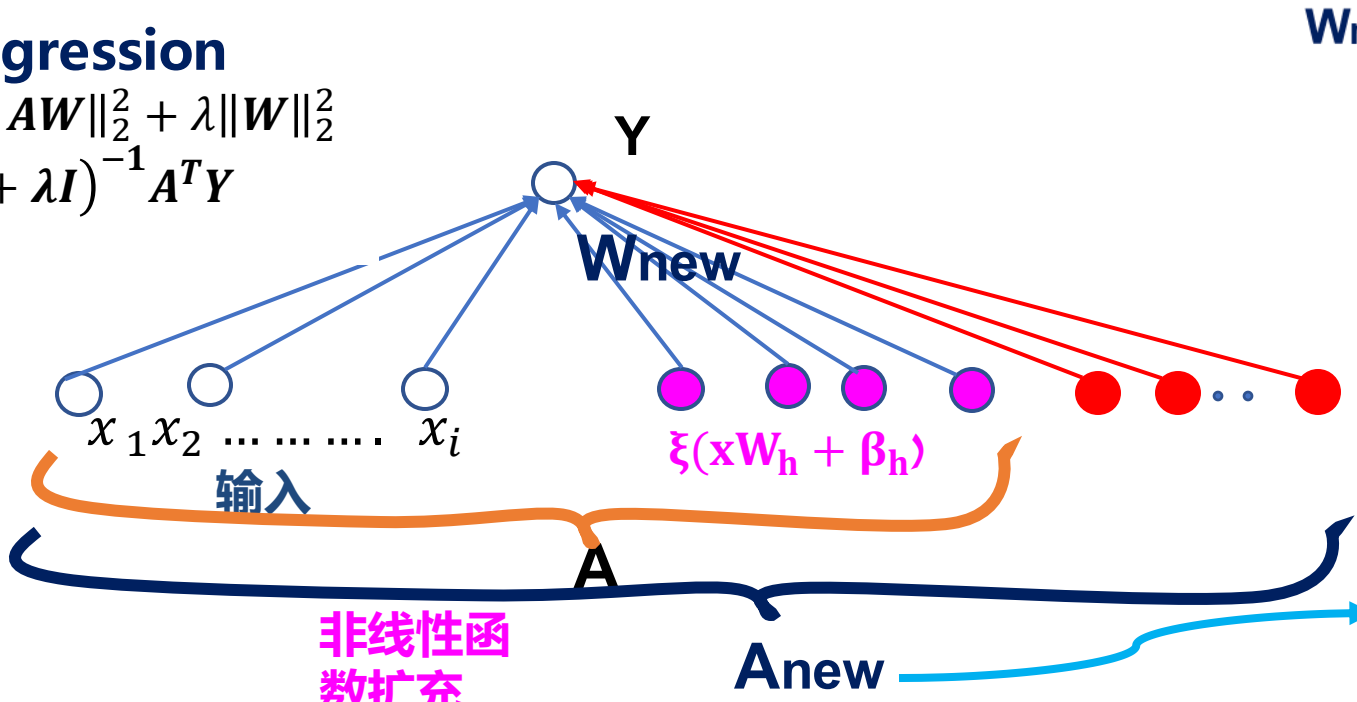


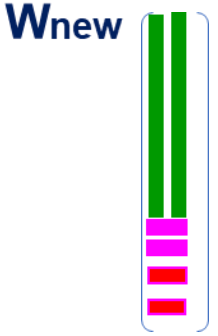
Learning—Reaching the accuracy: Dynamically adding neurons (expanding the structure) –-incremental learning

Least Square Regression

$$arg\ min_W: \|Y - AW\|_2^2 + \lambda\|W\|_2^2$$

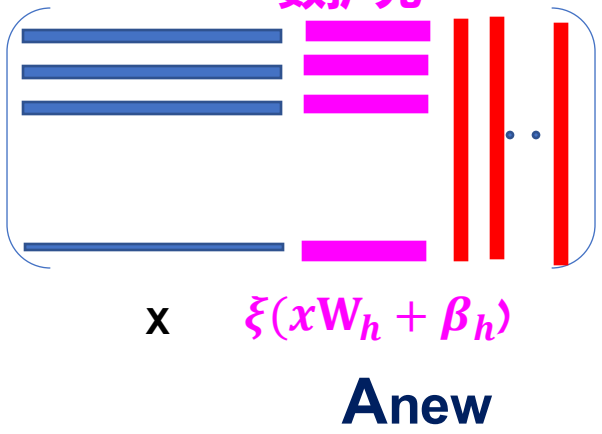
$$W = \left(A^T A + \lambda I\right)^{-1} A^T Y$$

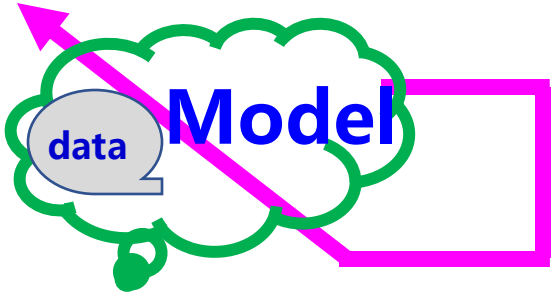




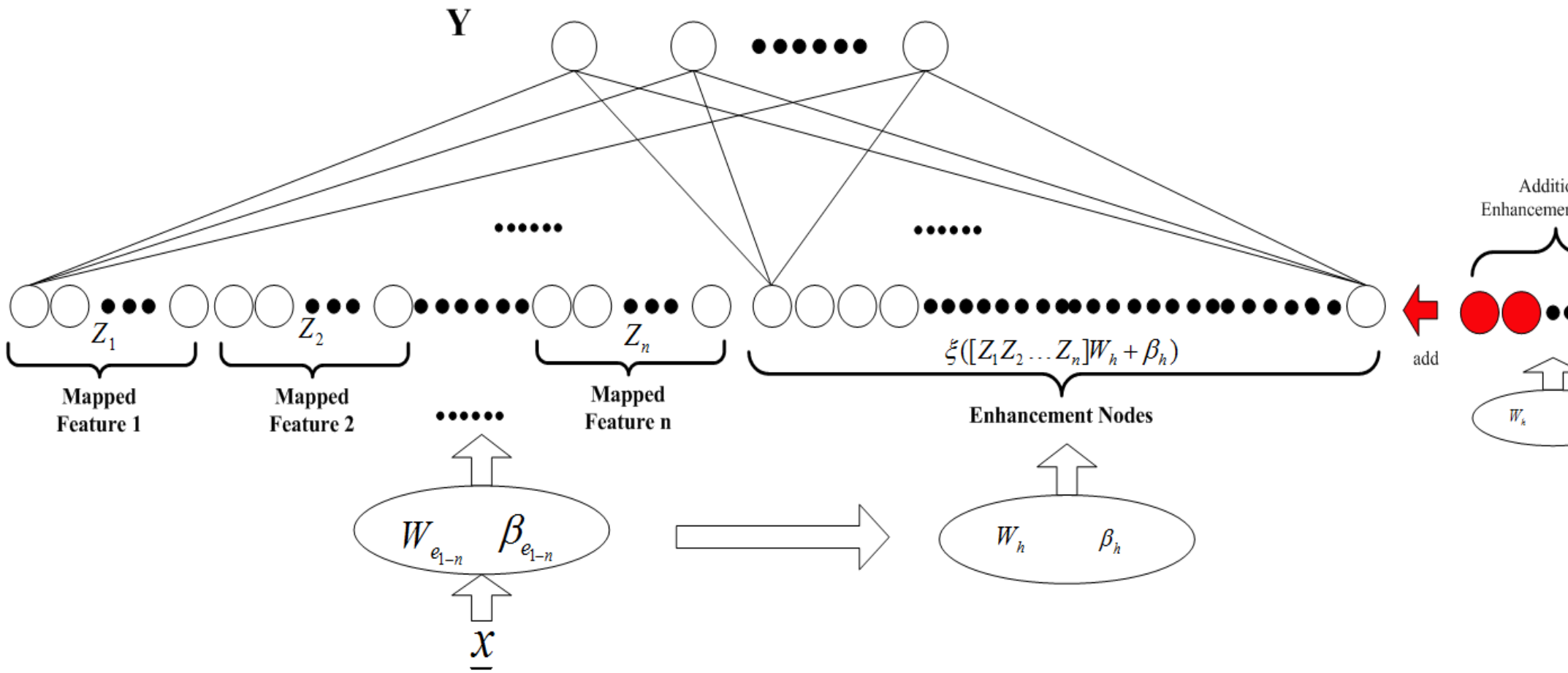
Solve AnewWnew = Y

$$arg\ min_{Wnew}: \|Y - Anew\,Wnew\|_2^2 + \lambda\|Wnew\|_2^2$$



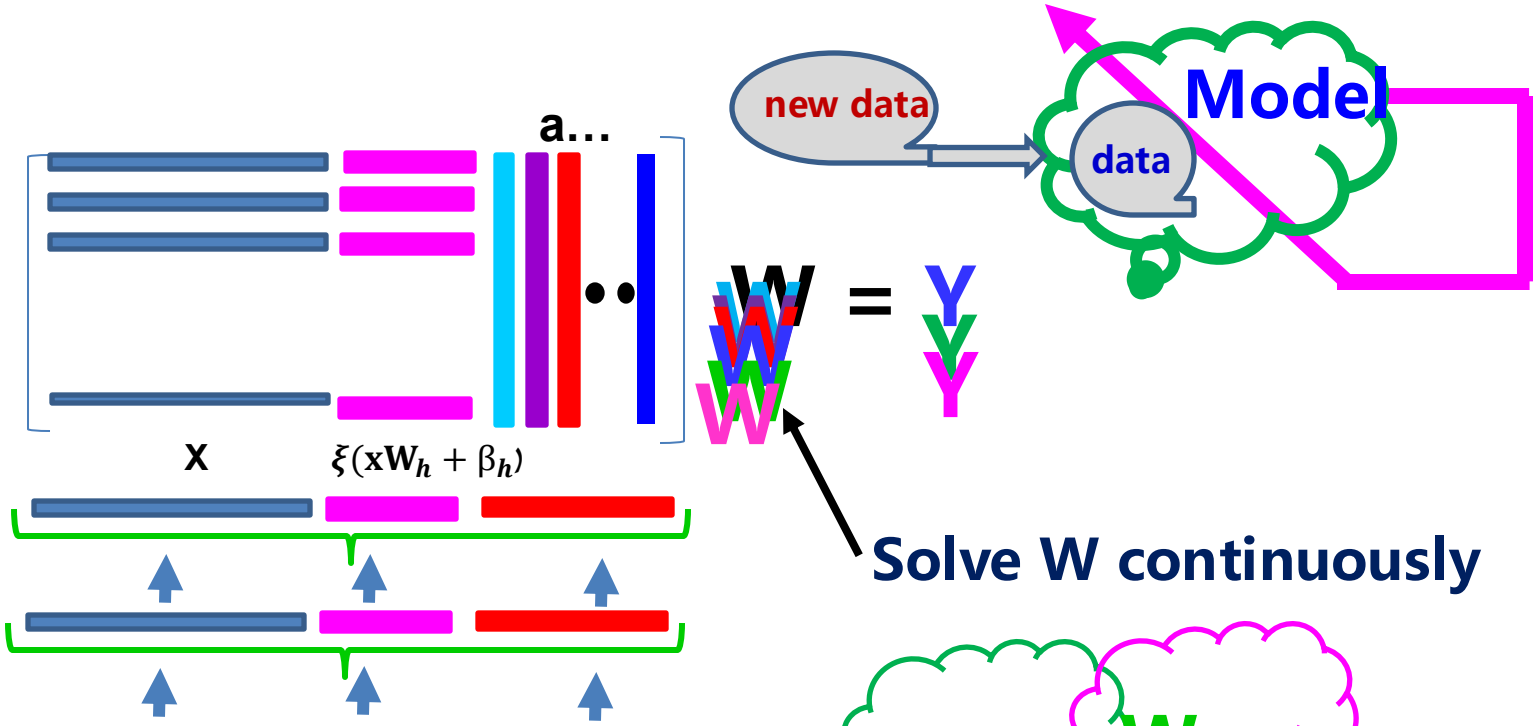


Picturing: Dynamically Adding Neurons: reach learning accuracy



- **CLP Chen** and ZL Liu, "Broad Learning System: An Effective and Efficient Incremental Learning System Without the Need for Deep Architecture," **IEEE Trans on Neural Networks and Learning Systems**, Vol, 29, No.1, pp. 10-24, **2018.**
- **CLP Chen,** ZL Liu, and S. Feng, "Universal approximation capability of broad learning system and its structural variations," 2018 (SMC), **," IEEE Trans on Neural Networks and Learning Systems**, Vol, 30, No.4, pp. 1191-12044, **2019.**

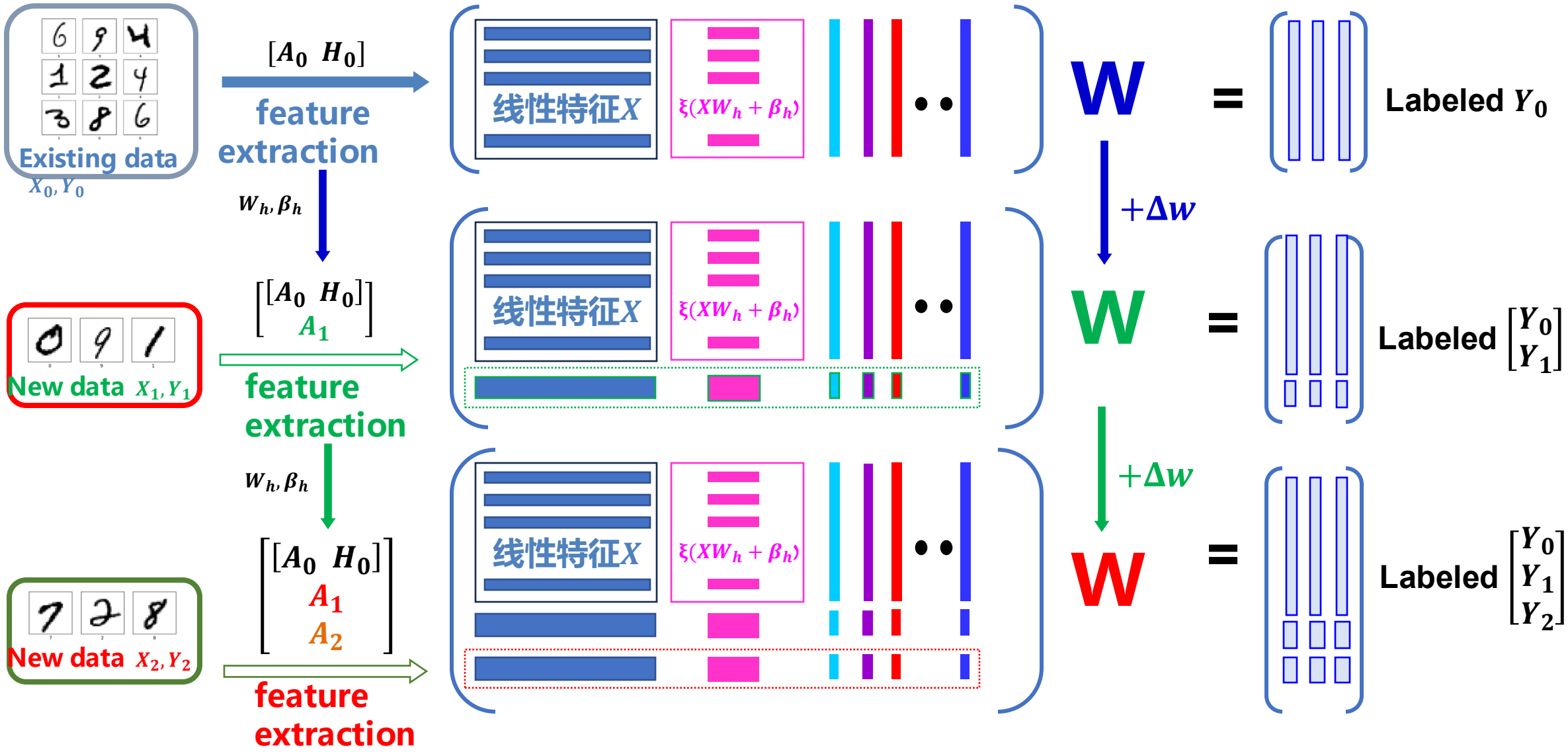
Adding new streaming data: dynamically incremental learning



Note: 当加入新数据时，W的维度是 列 的增加

Dimension of W increases columnwise-ly while adding the new data

Picturing: incrementally learning for new data stream: dynamically restructure the model



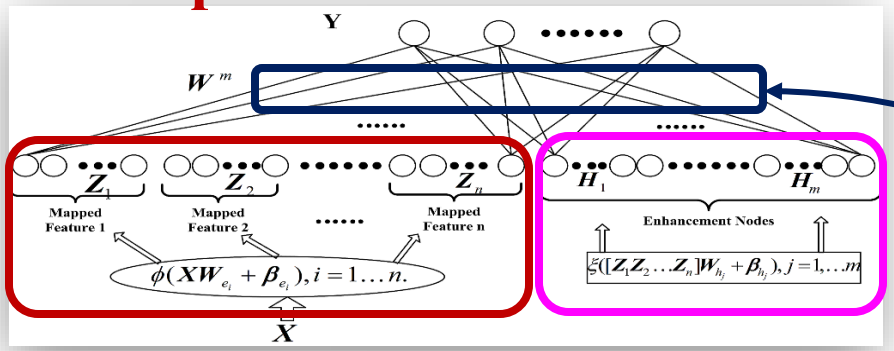
Broad Learning System

Invented by Philip Chen

Core Theory of BLS and Method: convert Regression to light weighted broad structure and can be computed in real-time

Machine learning is to solve Inhomogeneous non-linear equations AW=Y (i.e., BLS structure), Do not need deep structure learning

Core Modules in BLS



- **Random Sparse Autoencoder**

$$arg\,min_{\widehat{W}}: \left\|Z\widehat{W}-X\right\|_2^2+\sigma\left\|\widehat{W}\right\|_1 \quad (1)$$

- **Random Projection**

$$H_j=\xi_j\left(Z^nW_{h_j}+\beta_{h_j}\right), j=1,2,\ldots,m, \quad (2)$$

- **Feature Extension**

$$A=[Z^n,H^m]$$

- **Least Square Regression**

$$arg\,min_W: \|Y-AW\|_2^2+\lambda\|W\|_2^2 \quad (3)$$

$$W=\left(A^TA+\lambda I\right)^{-1}A^TY \quad (4)$$

Or by SGD Methods

Or by Greville's Method

$$\begin{cases} B=\begin{cases} C^+, & \text{if } C\neq 0, \\ A^+D\left(1+D^TD\right)^{-1}, & \text{if } C=0, \end{cases} \\ C=\tilde{A}-D^TA, \quad D^T=\tilde{A}A^+. \end{cases}$$

Finally, the updated weights $\hat{W}$ are analytically computed as:

$$\hat{W}=W+B(\tilde{Y}-\tilde{A}W).$$

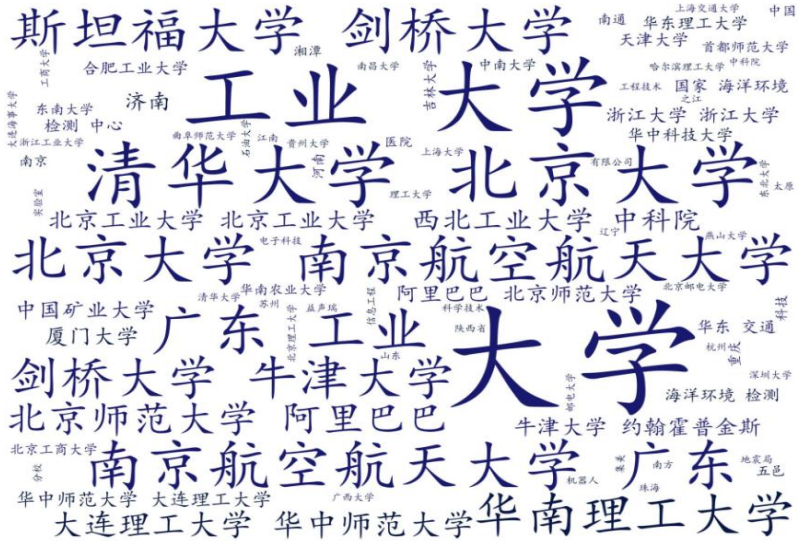
- **CLP Chen** and ZL Liu, "Broad Learning System: An Effective and Efficient Incremental Learning System Without the Need for Deep Architecture," **IEEE Trans on Neural Networks and Learning Systems**, Vol, 29, No.1, pp. 10-24, **2018. (影响因子: 11.683，Top期刊, 高被引論文。)**, Google學術引用2102次，Web of Science學術引用 1500+次。

D·SAIL 数据科学与人工智能研究团队

Broad Learning Systems development history

BLS is a shallow NN，It features efficient training and dynamic incremental growth. It is an efficient solution for lightweight deployment at the edge to achieve collaboration between large and small models.

domestic and international recognition: Stanford University、University of Oxford、Tsinghua University etc. 110+ research institutes，published 1500+ BLS related papers, more than 300 patents filed.



2017-2018

BLS invented

2018-2021

Fuzzy BLS
Unsupervised BLS
Semi-Supervised BLS
Stacked BLS
...

2022

Tree-shaped structure
Tree BLS
Imbalance
Double kernel BLS
Federated
FCL-BL
...

2023

Attention
Multi-Attn BLS
Twin networks
Twin-BLS
Graphed Networks
GEDBLS
...

2024

Continuous learning
BLS-CIL
QR decomposition
QR BLS
Recommendation system
BroadCF
...

2025

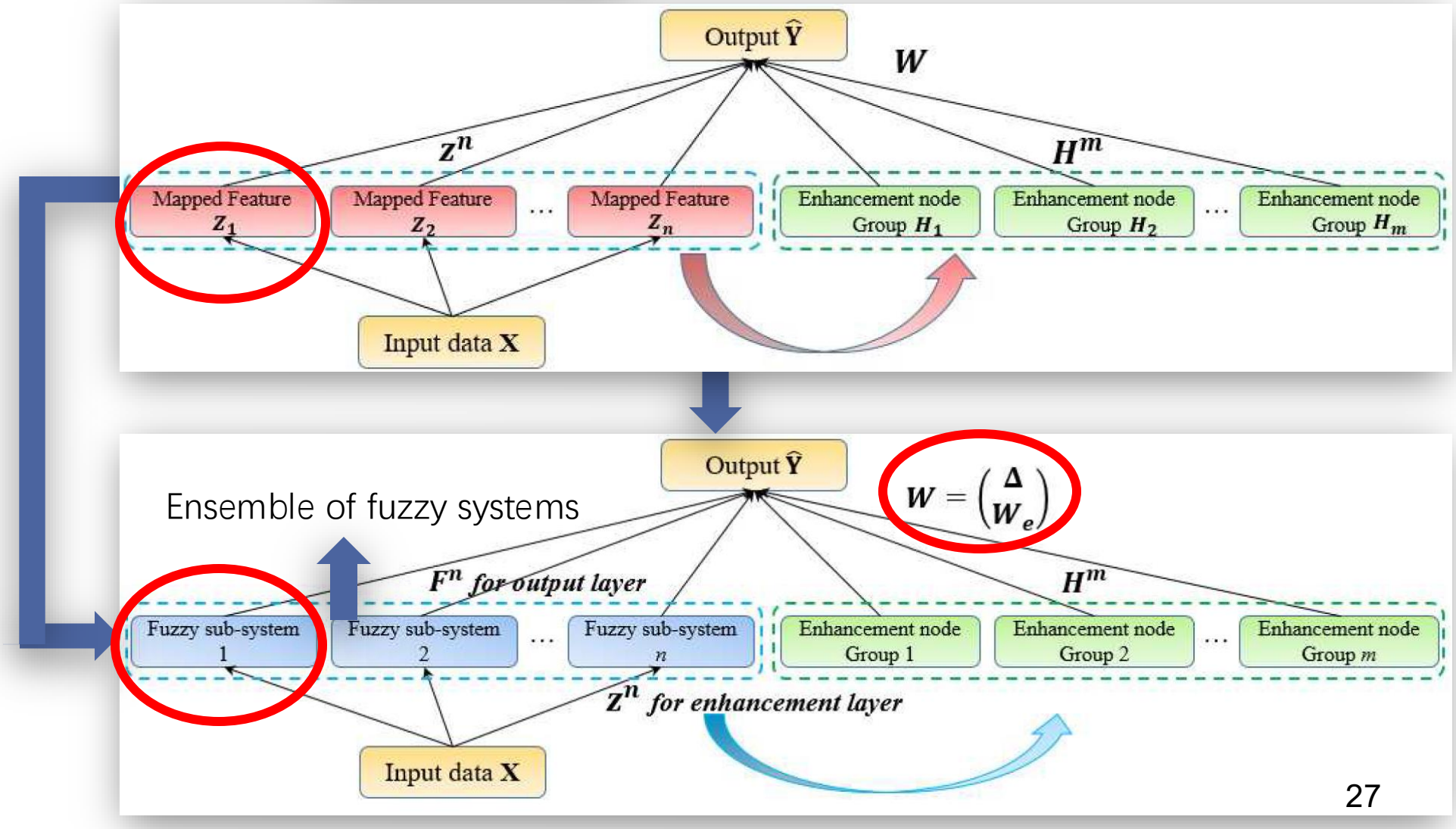
Online
OLBLS
Category incrementsG
CBLS
...

X. Gong, T. Zhang*, C. L. P. Chen and Z. Liu, "Research Review for Broad Learning System: Algorithms, Theory, and Applications," in IEEE Transactions on Cybernetics, doi: 10.1109/TCYB.2021.3061094.

2 Structure of a Fuzzy Broad Learning System

Broad Learning System → Fuzzy Broad Learning System

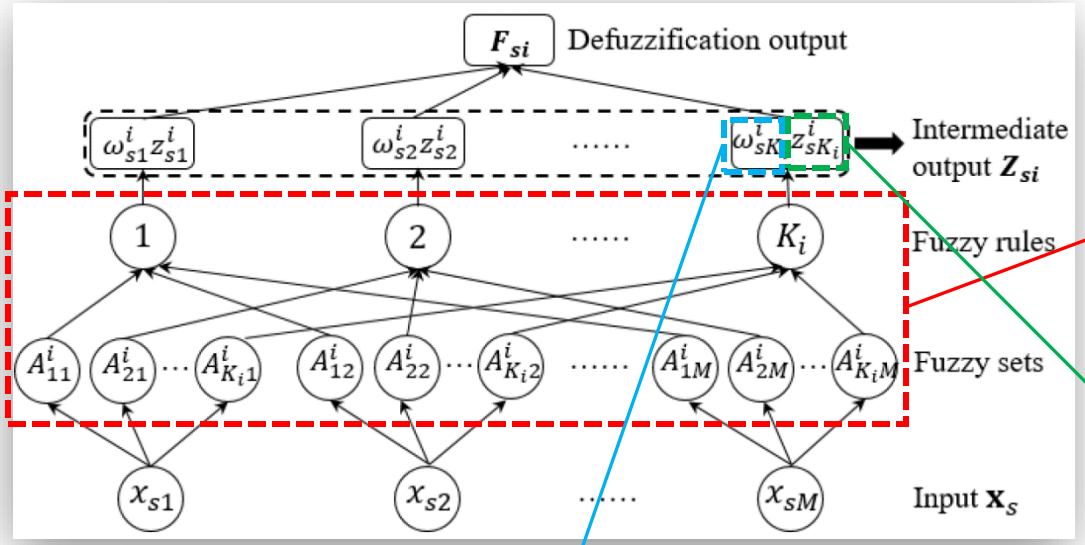
- Remove the sparse-autoencoder
- Replace the feature nodes with Takagi-Sugeno fuzzy sub-systems



2.1 Fuzzy BLS

Inputs for enhancement layer

Structure of the i-th fuzzy sub-system



Suppose that there are n fuzzy sub-systems and m groups of enhancement nodes in a Fuzzy BLS. The input data are $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N\times M}$ where $\boldsymbol{x}_s = (x_{s1}, x_{s2}, \ldots, x_{sM}), s = 1, 2, \ldots, N$. Suppose that there are K_i fuzzy rules in the ith fuzzy sub-system which have the following form

If x_{s1} is A_{k1}^i and x_{s2} is $A_{k2}^i \ldots$ and x_{sM} is A_{kM}^i
then $z_{sk}^i = f_k^i(x_{s1}, x_{s2}, \ldots, x_{sM}), k = 1, 2, \ldots, K_i.$

We adopt the first-order TS fuzzy system and let

$$z_{sk}^i = f_k^i(x_{s1}, x_{s2}, \ldots, x_{sM}) = \sum_{t=1}^{M} \alpha_{kt}^i x_{st}, \tag{7}$$

where α_{kt}^i is the coefficient.

The fire strength of the kth fuzzy rule in the ith fuzzy sub-system is

$$\tau_{sk}^i = \prod_{t=1}^{M} \mu_{kt}^i(x_{st}), \tag{8}$$

and we denote the weighted fire strength for each fuzzy rule as

$$\omega_{sk}^i = \frac{\tau_{sk}^i}{\sum^{K_i} \tau_{sk}^i}. \tag{9}$$

Fuzzy broad learning system: A novel neuro-fuzzy model for regression and classification, IEEE Transactions on Cybernetics 50 (2), 414-424,2018

Source codes link：

http://dsail.vip/bls.html

相关论文链接

- Universal approximation capability of broad learning system and its structural variations
- Research review for broad learning system: Algorithms, theory, and applications
- Stacked broad learning system: From incremental flatted structure to deep model

Source codes： ➡ **代码**

- BLS

SOME RECENT APPLICATIONS OF BLSS DISCUSSED IN SECTION IX

Field	Subfield	Method
Computer vision and image processing	Face recognition	Zhang et al. [144], P-BLS [145], ICBL [146] vision-brain [147], Zhang et al. [148], Luan et al. [149]
	Action recognition	DW-Net [136], Fisher-BLS [150], Sun et al. [151] Wavelet BLS [152], Lei et al. [153], Li et al. [154] FBLS-MD [155], Lin et al. [156], Luo et al. [157]
	Image classification	DABL [94], GBLS [93], Guo et al. [158], Cai et al. [159] AdaBoost-BLS [160], DBNet [161], Dang et al. [162] Distributed BLS [163], IWBLS [164], SS-BLS [115], UBL [125] BLSCF [165], Han et al. [166], EAF-BAP [167], Wang et al. [168]
	Image processing	Low-rank BLS [169], Guo et al. [83], DCBLS [107] Jin et al. [170], SID-BLS [84], Shrivastava et al. [171] SAE-BLS [172], Lei et al. [173], Chen et al. [174], Zhai et al. [175]
Medical/biomedical data analysis	Electroencephalography (EEG)	GCB-Net [137], CNBLS [176], GSS-BLS [118], TCBLS [177] MvBLS [87] Mixed-Norm BLS [178], BDGLS [168] LPVG-based BLS [179], CTW-BLS [180], Zou et al. [181]
	Medical data analysis	LPI-BLS [101], Luo et al. [182], MEKM-BLS [183] MISSIM [184], FNT [185], Chew et al. [186], CFCEBLS [187]
	Medical image segmentation	Liu et al. [188], Atlas-BLS [189]
System modeling and prediction	Time series prediction	MIE-BLS [190], R-BLS [64], SM-BLS [95], EMD-BLS [191] BLS-EC [105], Spatio-Temporal BLS [192], Zhou et al. [193] RM-BLS [96], SPCA-BLS [194], Yang et al. [195]
	System modeling	Weight BLS [77], CCBLS [196], Kuok et al. [197] ProBLS [198], Sei-BLS [199], RDBLS [200], BCRBM [201] Regularized-BLS [53], MRBL [202], Lai et al. [203]
Internet and communication engineering	Network and information security	RBF-BLS [204]–[206], Ma et al. [207] Zhong et al. [208], BL-TDA [209], ASSD [210]
	Internet and smart city	Wei et al. [211], BRL [212] Broad Forest [213], Xu et al. [214]
Control and robotics	Robotics	BLS-FDDEIF [215], BLS-SMC [216], BLS-FONTSM [217]
	System control	Adaptive-BLS [218], RIPF [219], BFNN [85] DT-BLS [220], Feng et al. [80], Yuan et al. [221] WBLAF [36], [39], QBLS [38], TDFW-BLS [37]
Fault dectect and diagnosis	Motors fault diagnosis	BCNN [74], PABSFD [222], bagging-BLS [104] Wang et al. [223], Jiang et al. [224], transfer-BLS [225] Multi-stage BLS [226], FIBL [227], Han et al. [228], OBLS [229]
Nature language processing	Text classification	Gate-BLS [63], BMT-Net [135], BroXLNet [138]

X. Gong, T. Zhang*, C. L. P. Chen and Z. Liu, "Research Review for Broad Learning System: Algorithms, Theory, and Applications," in IEEE Transactions on Cybernetics, Vol 52, Issue 9, pp. 8922-8950, 2021, doi: 10.1109/TCYB.2021.3061094.

信息科学领域位居前十位的热点前沿主要集中于人工智能基础理论方法、6G通信、人-机交互、类脑智能、医学信息处理等方向。人工智能基础理论方法方面的热点前沿有5个，生成式对抗网络、宽度学习系统、用于边缘计算的联邦学习成为新的热点前沿，可解释人工智能从去年的新兴前沿成为今年的热点前沿；强化学习相关前沿多次出现在热点前沿中，本期重点是推动强化学习解决真实世界问题的MuZero算法。6G通信方面的热点前沿有两个，深度学习在物理层通信中的应用首次成为热点前沿，可重构智能超表面是从去年的新兴前沿进入到今年的热点前沿。在人-机交互方面，下一代VR/AR实时全息近眼显示方法首次成为热点前沿。在类脑智能方面，脉冲神经网络及其神经形态芯片首次出现。在医学信息处理方面，用于脑电信号分析的卷积神经网络首次成为热点前沿。

表 52 信息科学领域 Top 10 热点前沿

排名	热点前沿	核心论文	被引频次	核心论文平均出版年
1	用于边缘计算的联邦学习	22	3682	2020.2
2	宽度学习系统	6	1053	2020.0
3	可重构智能超表面	32	9372	2019.7
4	下一代 VR/AR 实时全息近眼显示方法	3	457	2019.3
5	可解释人工智能	4	2900	2019.0
6	脉冲神经网络及其神经形态芯片	13	2931	2018.6
7	深度学习在物理层通信中的应用	13	2949	2018.5
8	生成式对抗网络	8	15051	2018.4
9	MuZero 强化学习算法	6	3607	2018.3
10	用于脑电信号分析的卷积神经网络	9	2531	2018.2

喜报

2023年11月28日，中国科学院科技战略咨询研究院、中国科学院文献情报中心与科睿唯安联合发布的《2023研究前沿》报告和《2023研究前沿热度指数》报告，遴选出2023年全球较为活跃或发展迅速的128个研究前沿，并对相关学科的发展趋势和重点问题进行了研判。

中国自动化学会副理事长、华南理工大学教授陈俊龙首创的"宽度学习系统"，荣获2023年信息科学领域热点前沿第二名！

Feature increments

Data increment

Wu WenJun AI Outstanding Contribution Award, China

表 52 信息科学领域 Top 10 热点前沿

排名	热点前沿	核心论文	被引频次	核心论文平均出版年
1	用于边缘计算的联邦学习	22	3682	2020.2
2	宽度学习系统	6	1053	2020.0
3	可重构智能超表面	32	9372	2019.7
4	下一代 VR/AR 实时全息近眼显示方法	3	457	2019.3
5	可解释人工智能	4	2900	2019.0
6	脉冲神经网络及其神经形态芯片	13	2931	2018.6
7	深度学习在物理层通信中的应用	13	2949	2018.5
8	生成式对抗网络	8	15051	2018.4
9	MuZero 强化学习算法	6	3607	2018.3
10	用于脑电信号分析的卷积神经网络	9	2531	2018.2

首提"宽度学习系统"，实现边缘端智能学习

Dynamic deep neural networks (Based on incremental stacked BLS)

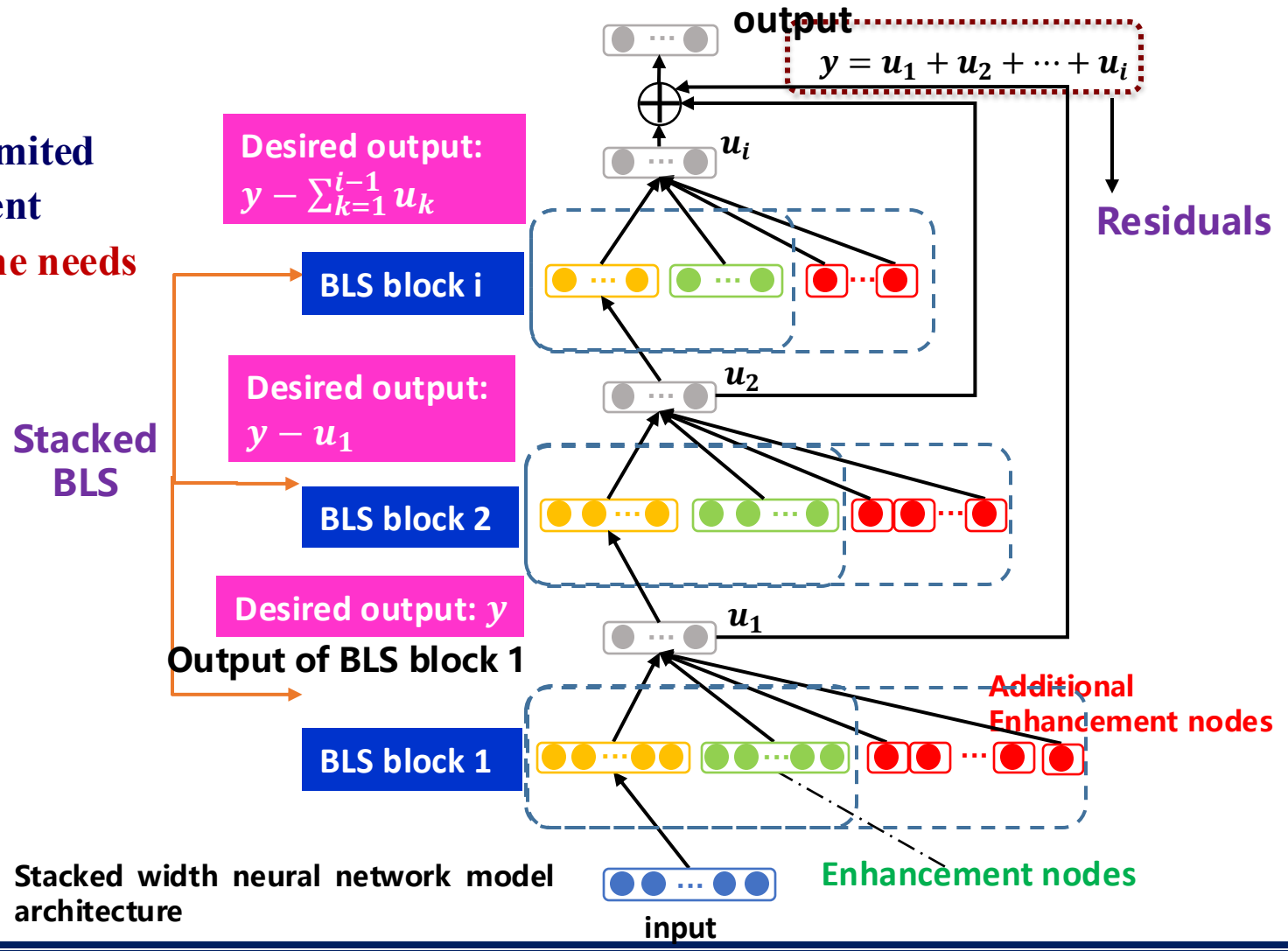
Challenge

Currently, network structures have limited precision, fixed frameworks, and insufficient learning capabilities. This does not meet the needs of brain-inspired research, and innovative computing frameworks need to be further designed based on brain-related models.

Innovation

Stacking multiple BLS blocks forms a deep network structure.

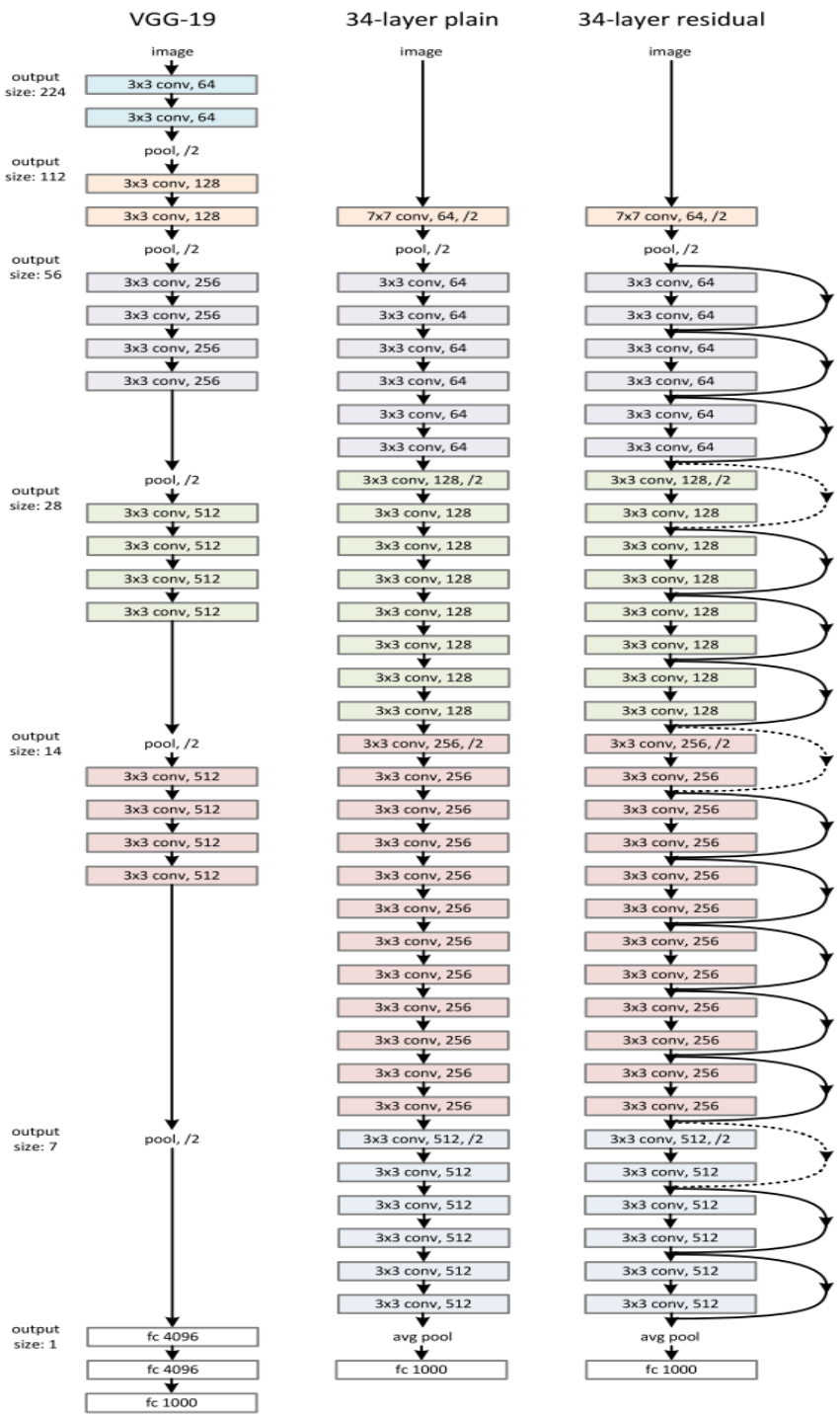
Introducing the concept of residuals，。



Stacked width neural network model architecture

[1] Liu, Zhulin, CL Philip Chen, Shuang Feng, Qiying Feng, and Tong Zhang. "Stacked broad learning system: From incremental flatted structure to deep model." IEEE Transactions on Systems, Man, and Cybernetics: Systems 51, no. 1 (2020): 209-222. (SCI一区TOP，IF：13.451)

➢ ResNet


VGG-19
34-layer plain
34-layer residual
image
output size: 224
3x3 conv, 64
3x3 conv, 64
pool, /2
output size: 112
3x3 conv, 128
3x3 conv, 128
pool, /2
output size: 56
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
pool, /2
output size: 28
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
pool, /2
output size: 14
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
pool, /2
output size: 7
fc 4096
fc 4096
fc 1000
output size: 1
7x7 conv, 64, /2
pool, /2
3x3 conv, 64
3x3 conv, 128, /2
3x3 conv, 128
3x3 conv, 256, /2
3x3 conv, 256
3x3 conv, 512, /2
3x3 conv, 512
avg pool
fc 1000

Compared with ResNet Structure

TABLE XII: Performance comparison between Stacked CCFBLS and ResNet in SVHN

Algorithms	ResNet-50	CCFBLS	Stacked CCFBLS
Testing Accuracy	92.06%	96.69%	**97.12%**
No. of blocks	-	-	4
No. of layers	50	26	35
No. of Param.	23.521M	2.934M	2.959M

Fig. 12: Example images selected from SVHN Dataset

TABLE XI: Performance comparison between Stacked CCFBLS and ResNet in CIFAR-10

Algorithms	ResNet-32	CCFBLS	Stacked CCFBLS
Testing Accuracy	92.49%	94.29%	**94.78%**
No. of blocks	-	-	3
No. of layers	32	26	32
No. of Param.	11.174M	2.933M	2.951M

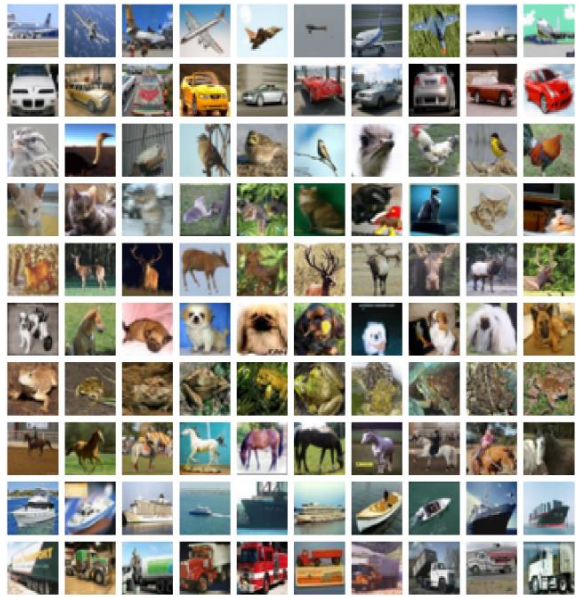


Fig. 11: Example images selected from CIFAR-10 Dataset

Compared with ResNet Structure

TABLE XIII
PERFORMANCE COMPARISON BETWEEN STACKED CCFBLS AND RESNET IN CIFAR100

Algorithms	ResNet-50	CCFBLS	Stacked CCFBLS
Testing Accuracy	78.51%	81.89%	**88.56%**
No. of blocks	-	-	3
Accumulative No. of layers	50	28	32
No. of Param.	23.706M	3.658M	3.803M

TABLE XIV
PERFORMANCE COMPARISON WITH SELECTED STATE-OF-THE-ART DEEP METHODS

Algorithms	CIFAR-10	CIFAR-100	SVHN
Capsule Net [56]	89.3%	-	95.7%
GDCNN [57]	89.23%	66.7%	-
Meta Net [58]	88%	-	-
CGap [59]	93.59%	73%	96.25%
SReluDCNN [60]	93.02%	70.9%	-
Stacked CCFBLS	**94.78%**	**88.56%**	**97.12%**

脑认知的启发：正电子发射断层扫描脑成像（PET）

Brain Advance Access published July 14, 2016

doi:10.1093/brain/aww143

BRAIN 2016: Page 1 of 15 | 1

BRAIN
A JOURNAL OF NEUROLOGY

Aside: Robotic-Manipulator Control (advantage of Incremental Learning)

Motor Learning and Generalization Using Broad Learning Adaptive Neural Control

Haohui Huang, Tong Zhang, *Member, IEEE*, Chenguang Yang, *Senior Member, IEEE*, and C. L. Philip Chen, *Fellow, IEEE*

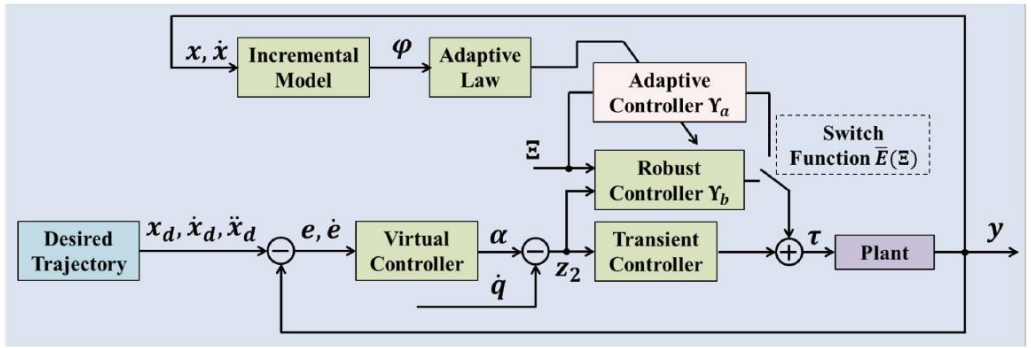


Fig. 2. Control diagram of an adaptive neural controller using a BLS.

Optimal Robot–Environment Interaction Under Broad Fuzzy Neural Adaptive Control

Haohui Huang, *Member, IEEE*, Chenguang Yang, *Senior Member, IEEE*, and C. L. Philip Chen, *Fellow, IEEE*

3826

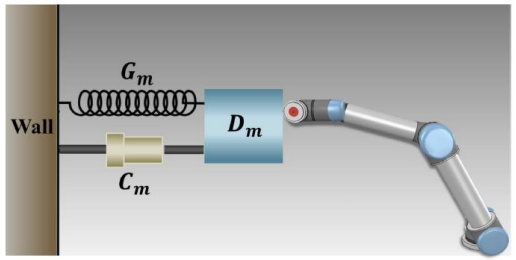


Fig. 1. Impedance model of the robot–environment interaction.

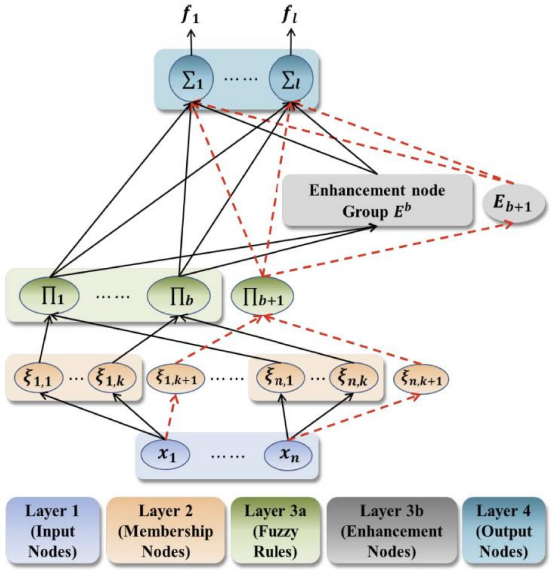
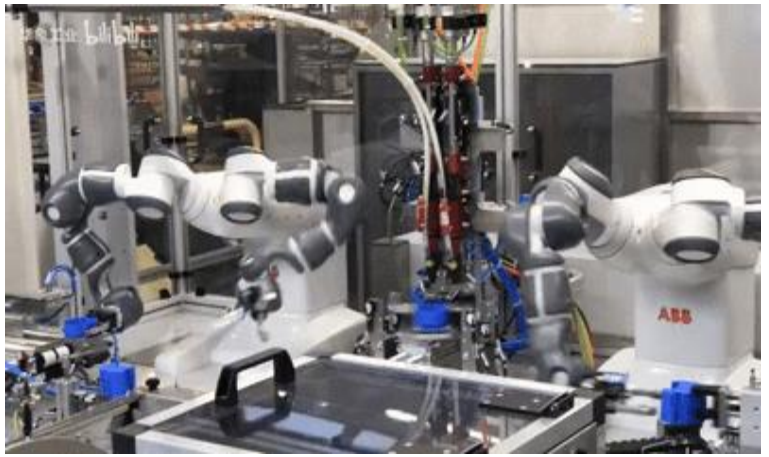
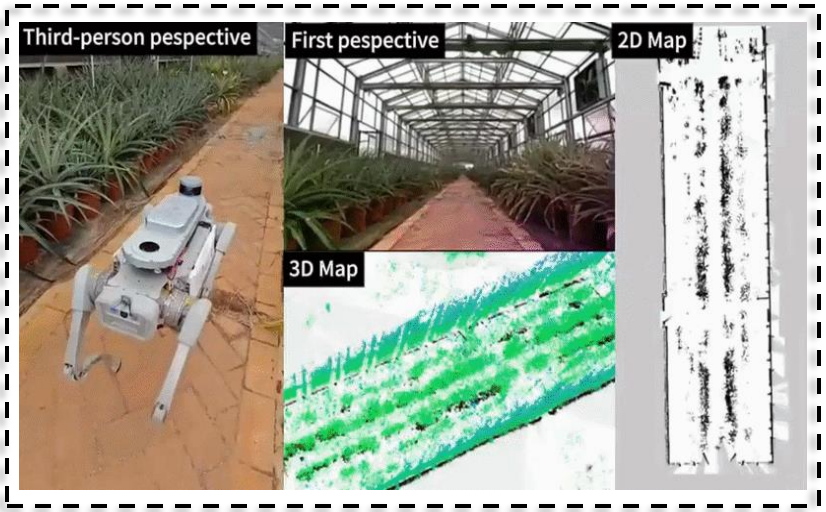
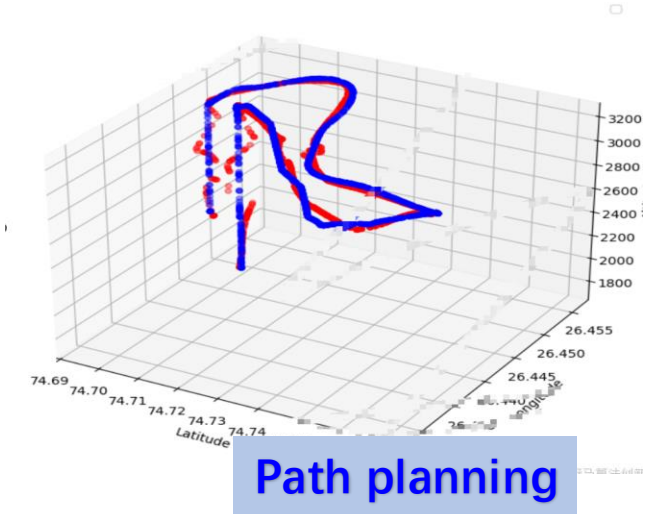


Fig. 4. Schematic of BFNN.

Unmanned System Applications：Incremental Learning in real-time



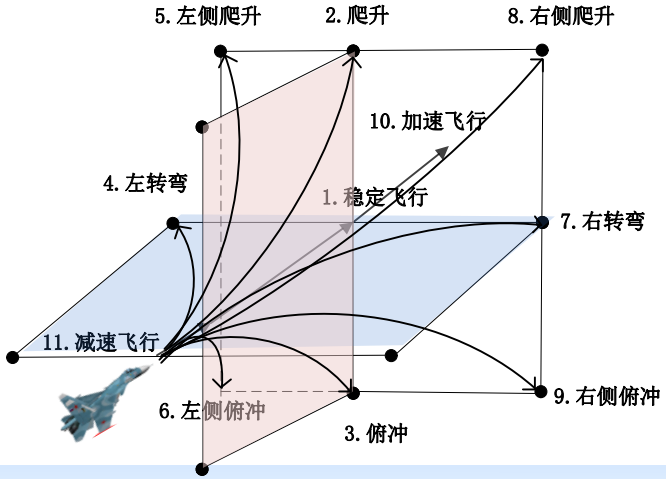




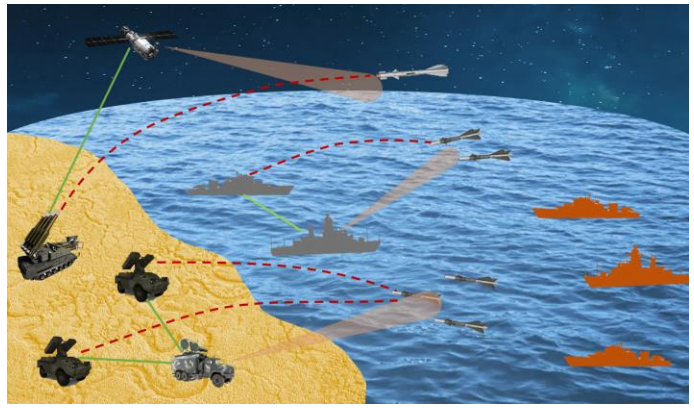
Path planning

Multi-aircraft adversarial behavior analysis based on a BLS is a key to victory in multi-aircraft combat in complex dynamic environments

Trajectory forecasting



Motion recognition



Intent to reason

Meaning and practical value

- ✓ Anticipate enemy actions in advance
- ✓ Improves attack accuracy
- ✓ Optimize tactical layout

- ✓ Identify false moves and confusing tactics
- ✓ Early warning and avoidance

- ✓ Psychological tactics and psychological advantage
- ✓ Optimize tactical decision-making
- ✓ Auxiliary intelligence and decision support

Inverse-free Broad Learning Systems

Previously, "pseudo-inverse" computation is needed

- **Least Square Regression**

$$arg\ min_{\boldsymbol{W}}: \|\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{W}\|_2^2+\lambda\|\boldsymbol{W}\|_2^2$$

$$\boldsymbol{W}=\left(\boldsymbol{A}^T\boldsymbol{A}+\boldsymbol{\lambda}\boldsymbol{I}\right)^{-1}\boldsymbol{A}^T\boldsymbol{Y}$$

Greville's Method

$$\begin{cases}\mathbf{B}=\begin{cases}\mathbf{C}^{+}, & \text{if } \mathbf{C}\neq 0,\\ \mathbf{A}^{+}\mathbf{D}\left(1+\mathbf{D}^{\mathrm{T}}\mathbf{D}\right)^{-1}, & \text{if } \mathbf{C}=0,\end{cases}\\ \mathbf{C}=\tilde{\mathbf{A}}-\mathbf{D}^{\mathrm{T}}\mathbf{A}, \quad \mathbf{D}^{\mathrm{T}}=\tilde{\mathbf{A}}\mathbf{A}^{+}.\end{cases}$$

其中 $D=(A^m)^{+}\xi(Z^nW_{h_{m+1}}+\beta_{h_{m+1}})$,

Finally, the updated weights $\hat{\mathbf{W}}$ are analytically computed as:

$$\hat{\mathbf{W}}=\mathbf{W}+\mathbf{B}\left(\tilde{\mathbf{Y}}-\tilde{\mathbf{A}}\mathbf{W}\right).$$

All improvements and various transformations are placed on the pseudo-inverse operation of solving W!

Inverse-free BLS

➢ **Least Square Regression**

$$arg\ min_{W}: \|Y - AW\|_2^2 + \lambda\|W\|_2^2$$

$$W = \left(A^T A + \lambda I\right)^{-1} A^T Y$$

Problem: inverse operations causes unstable values

Traditional analytical solution solving process

LSE

$$\min_{W_0} \|A_0 W_0 - Y_0\|_F$$

→ **semi-regular equation**

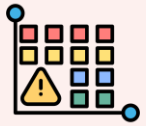
$$A_0^T A_0 W_0 = A_0^T Y_0$$

→ **Solution**

$$W_0 = (A_0^T A_0)^{-1} A_0^T Y_0$$

Need Inverse $(A_0^T A_0)^{-1}$ Large matrix、High Dimension– Singularity

Data Unstable



求解结果对数值敏感、波动大

High computing costs

高维矩阵求逆复杂度高

Real-time capability is limited

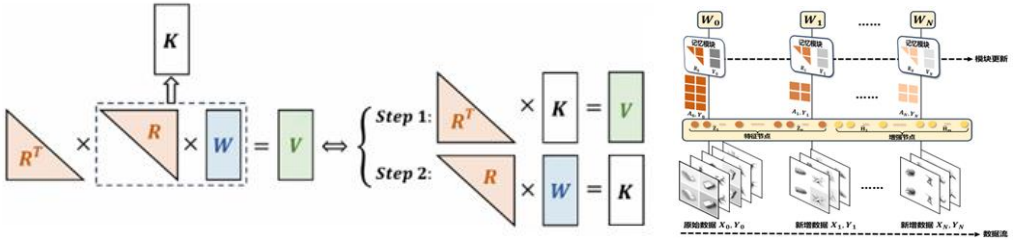
高维矩阵求逆计算时间长

去除求逆运算

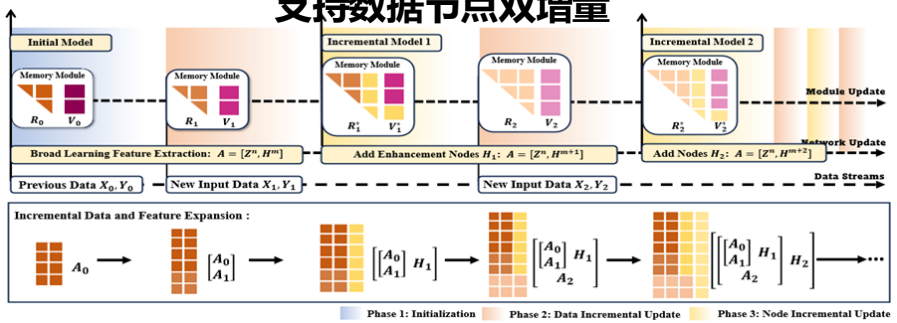
Inverse-free improvements: stable and efficient modeling

将最小二乘问题转化为**半正规方程**求解，借助**三角矩阵迭代计算**，**快速稳定**地解析计算更新权重。

$$\min_{W_0} \|A_0 W_0 - Y_0\|_F \rightarrow R_0^T R_0 W_0 = A_0^T Y_0 \triangleq V_0$$



支持数据节点双增量



对比伪逆方法准确率提升最高**2.56%**，计算速度提升**330.24%**!

无逆宽度学习通过**消除求逆依赖**，实现更**稳定、高效的协同建模**，更适用于**开放复杂环境下云边端智能的实时计算**。

Efficient incremental learning for Inverse Matrix-Free broad learning system, *GZ Chen, CY Lei, ZL Liu, CLP Chen, HW Sun*, **Information Fusion**, 127 (2026) 103842.

Efficient and high-precision incremental calculations： Single round

Inverse-free BLS is a breakthrough of data unstable, and achieve、parallelism and lightweighted computing

A paradigm for calculating data increments without inverted weights

- **The LSE problem is transformed into a semi-normalizagtion equation**

$$\min_{\mathbf{W}_0} \|\mathbf{A}_0\mathbf{W}_0 - \mathbf{Y}_0\|_F \rightarrow$$

To Solve W0 , multiply $\mathbf{A}_0^{\mathrm{T}}$ become $\mathbf{A}_0^{\mathrm{T}}\mathbf{A}_0$, become $\mathbf{R}_0^{\mathrm{T}}\mathbf{R}_0$

For large and dense $\mathbf{A}_0$, previous inverse-based incremental BLS explicitly computes analytical solution $\mathbf{W}_0$ by the pseudo-inverse $\mathbf{A}_0^{+}$ for initialization:

$$\mathbf{W}_0 = \mathbf{A}_0^{+}\mathbf{Y}_0 = \left(\mathbf{A}_0^{\mathrm{T}}\mathbf{A}_0 + \lambda\mathbf{I}\right)^{-1}\mathbf{A}_0^{\mathrm{T}}\mathbf{Y}_0,$$

while storing or updating the pseudo-inverse $\mathbf{A}_0^{+}$ is inefficient. In contrast, InvF-BLS overcomes these issues via reduced QR decomposition:

$$\mathbf{A}_0 = \mathbf{Q}_0\mathbf{R}_0,$$

where $\mathbf{Q}_0 \in \mathbb{R}^{l_0 \times p}$ is an orthogonal matrix and $\mathbf{R}_0 \in \mathbb{R}^{p \times p}$ is an upper triangular matrix.

This transforms the least squares problem into seminormal equations (SNE)[33], which is an easier-solving triangular system:

$$\mathbf{A}_0^{\mathrm{T}}\mathbf{A}_0\mathbf{W}_0 = \mathbf{R}_0^{\mathrm{T}}\underbrace{\mathbf{Q}_0^{\mathrm{T}}\mathbf{Q}_0}_{=\mathbf{I}}\mathbf{R}_0\mathbf{W}_0 = \mathbf{R}_0^{\mathrm{T}}\mathbf{R}_0\mathbf{W}_0 = \mathbf{A}_0^{\mathrm{T}}\mathbf{Y}_0 \triangleq \mathbf{V}_0. \tag{4}$$

And we set the triangular coefficient matrix and updated labels $(\mathbf{R}_0, \mathbf{V}_0)$ as the memory module to integrate the new information.

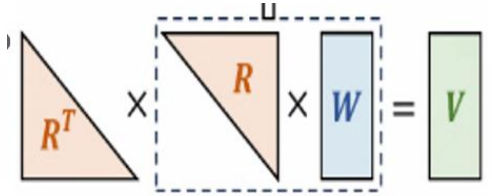
Efficient and high-precision incremental calculations： Single round

Inverse-free BLS is a breakthrough of data unstable, and achieve、parallelism and lightweighted computing

A paradigm for calculating data increments without inverted weights

- **The LSE problem is transformed into a semi-normalizagtion equation**

$$\min_{\mathbf{W}_0} \|\mathbf{A}_0\mathbf{W}_0 - \mathbf{Y}_0\|_F \rightarrow \mathbf{R}_0^{\mathrm{T}}\underbrace{\mathbf{R}_0\mathbf{W}_0}_{\boldsymbol{K}} = \mathbf{A}_0^{\mathrm{T}}\mathbf{Y}_0 \triangleq \mathbf{V}_0$$



Solve $\mathbf{R}_0^{\mathrm{T}}\boldsymbol{K} = \mathbf{V}_0$ find $\boldsymbol{K}$

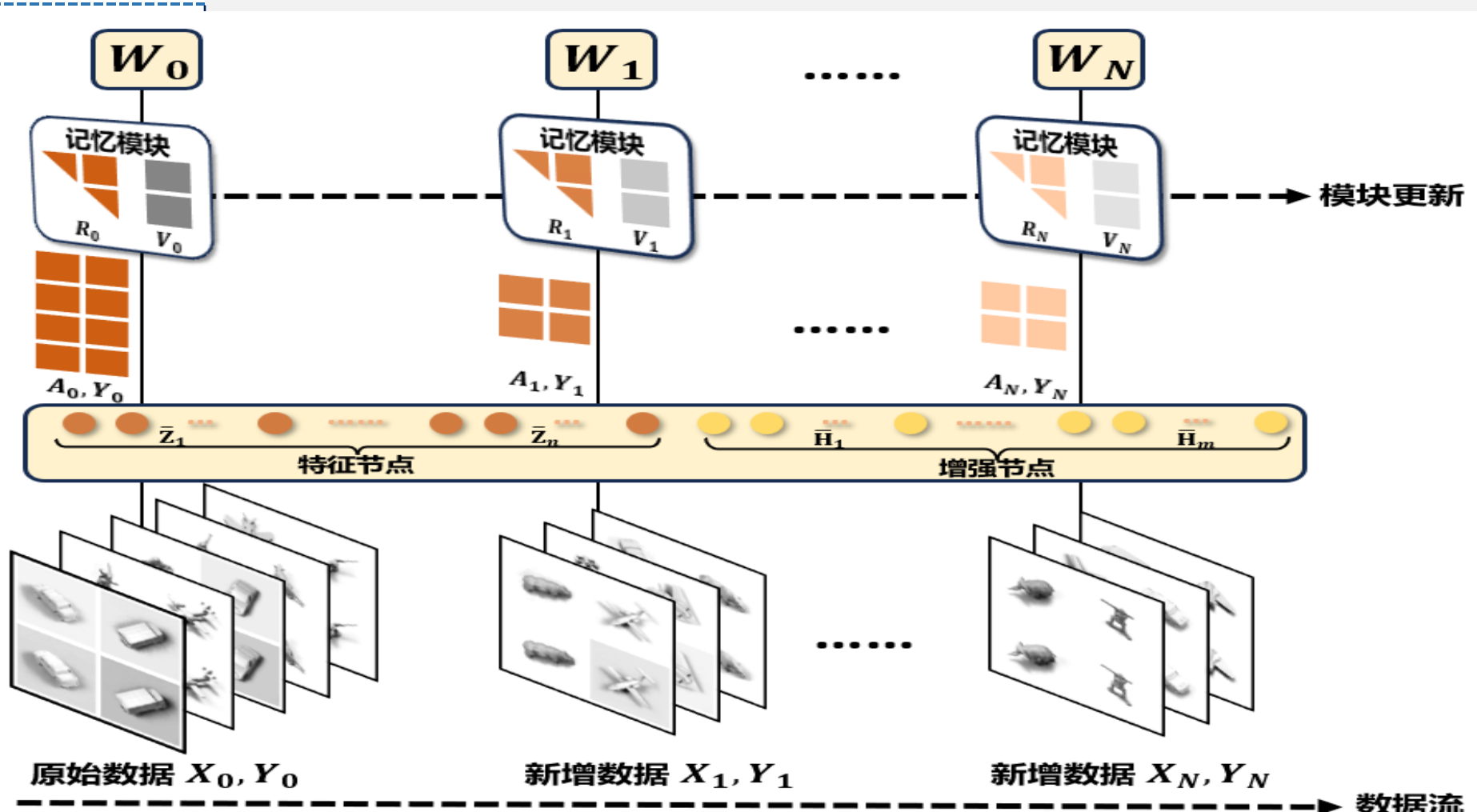
Then solve $\mathbf{R}_0\mathbf{W}_0 = \boldsymbol{K}$ find $\mathbf{W}_0$

Efficient and high-precision incremental calculations：Weight updated

A paradigm for calculating data increments without inverted weights

- achieve、parallelism 、memory module update algorithm adapted to data increments

- By using the triangular matrix for forward and backward calculation, the weights are quickly and stably parsed and updated



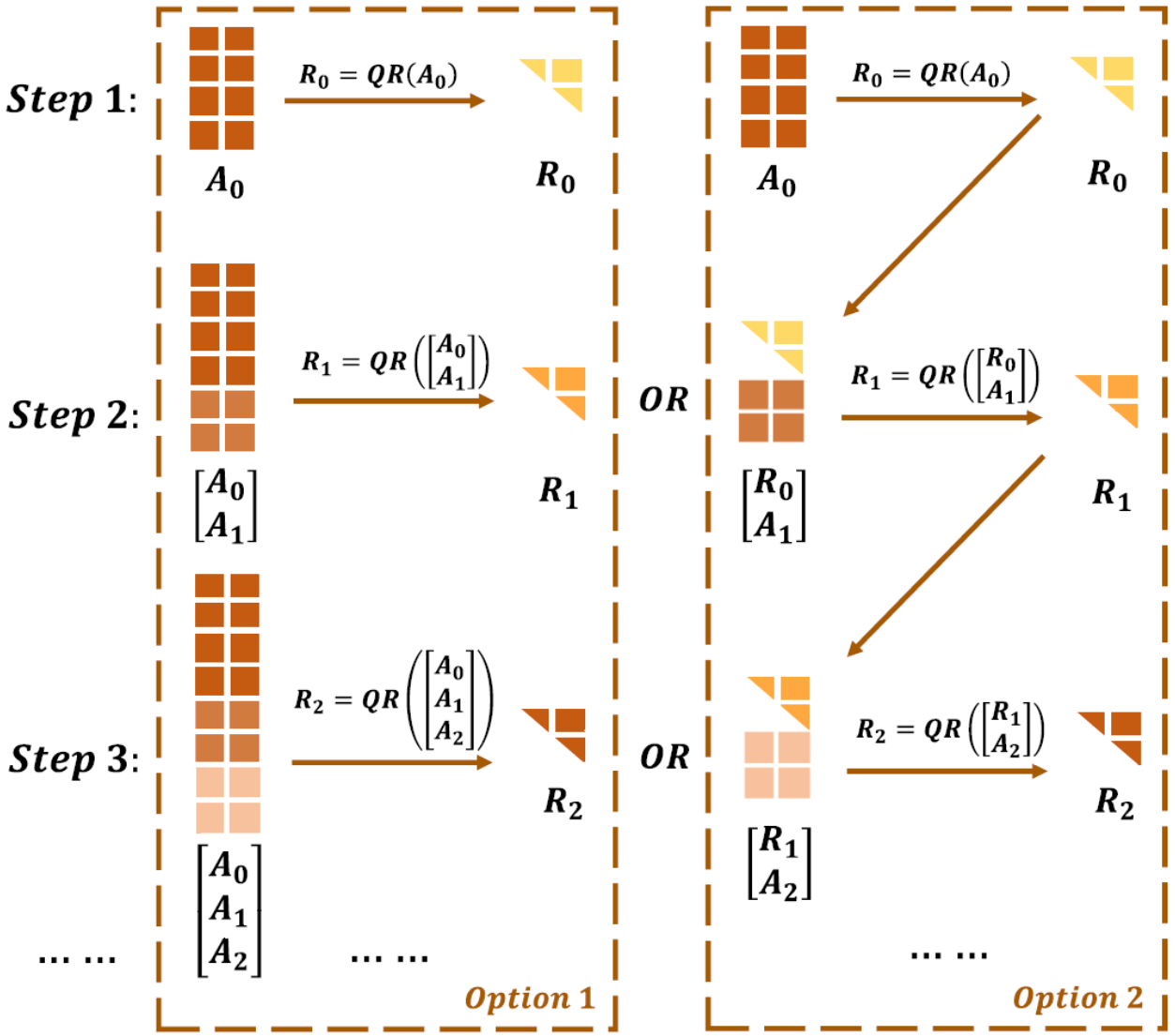


Fig. 4. Incremental R-factor update.

$$\mathbf{W}_k = \mathcal{BS}\left(\mathbf{R}_k, \mathcal{FS}\left(\mathbf{R}_k^{\mathrm{T}}, \mathbf{V}_k\right)\right),$$

where:

$$\begin{cases} \mathbf{R}_k = \mathcal{QR}\left(\begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix}\right), \\ \mathbf{V}_k = \mathbf{R}_{k-1}^{\mathrm{T}} \mathbf{R}_{k-1} \mathbf{W}_{k-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{Y}_k = \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{Y}_k. \end{cases}$$

This process enables efficient adaptation to recursive update chain:

$$\mathbf{R}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (7)}} \mathbf{R}_k \quad \text{and} \quad \mathbf{V}_{k-1} \xrightarrow{\text{Update in}}$$

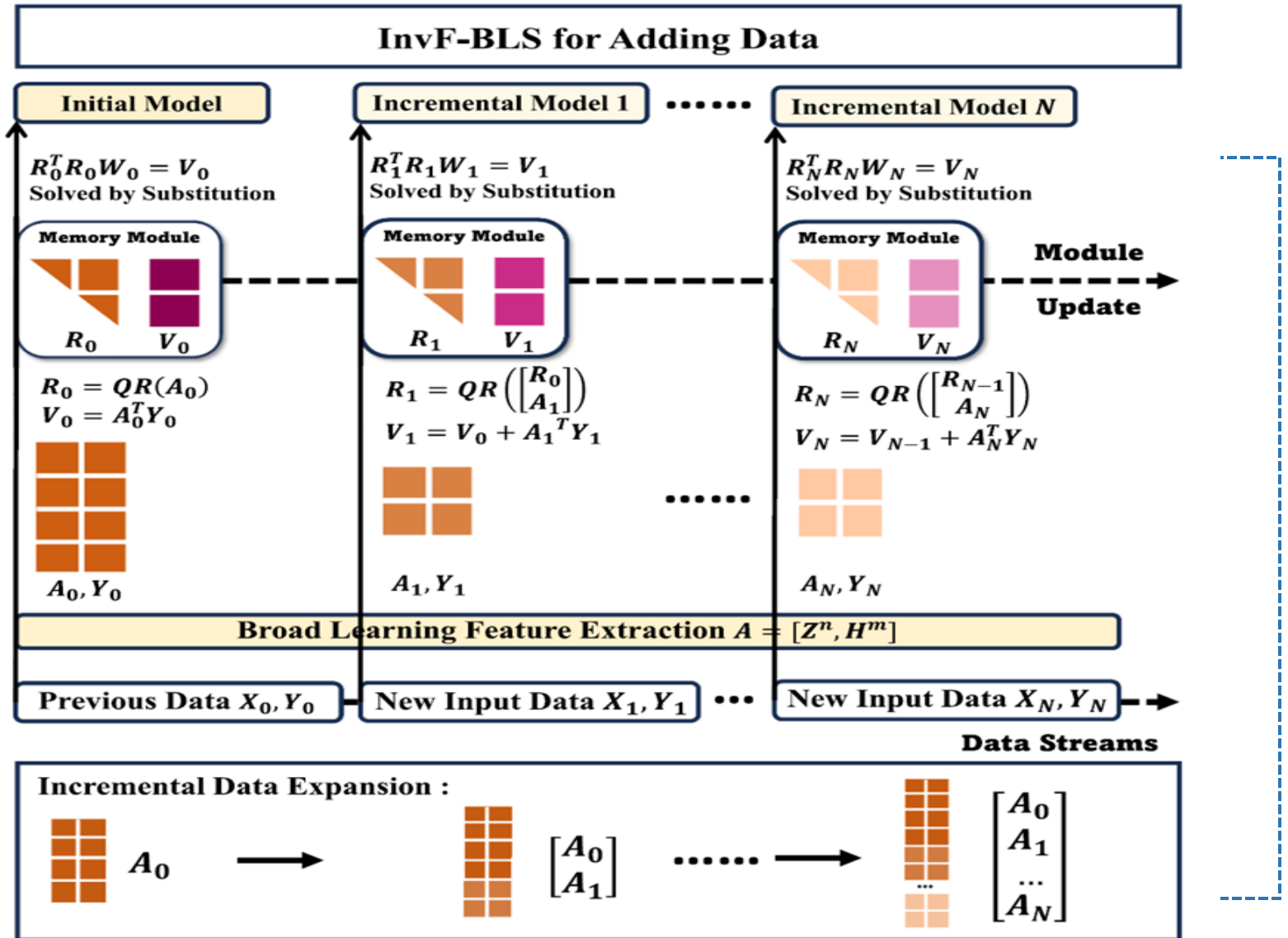
$$\Rightarrow \mathbf{W}_{k-1} \xrightarrow{\text{Update in(9)}} \mathbf{W}_k.$$

The overall process of InvF-BLS for adding data is illustrat and Algorithm 3.

Efficient incremental calculations in parallelism：Real-time data flow incremental calculation

Data increments

- Through progressive method, a memory matrix update algorithm is designed to support data increment and adapt to distributed computing needs under dynamic input.



Summary: Efficient and high-precision data incremental calculation：

$$\mathbf{W}_k = \mathcal{BS}\left(\mathbf{R}_k, \mathcal{FS}\left(\mathbf{R}_k^{\mathrm{T}}, \mathbf{V}_k\right)\right),$$

$$\begin{cases} \mathbf{R}_k = \mathcal{QR}\left(\begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix}\right), \\ \mathbf{V}_k = \mathbf{R}_{k-1}^{\mathrm{T}}\mathbf{R}_{k-1}\mathbf{W}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k = \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k. \end{cases}$$

This process enables efficient adaptation to streaming data by the recursive update chain:

$$\mathbf{R}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (7)}} \mathbf{R}_k \quad \text{and} \quad \mathbf{V}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in(8)}} \mathbf{V}_k$$

$$\Rightarrow \mathbf{W}_{k-1} \xrightarrow{\text{Update in(9)}} \mathbf{W}_k.$$

Real-time distributed Data an Nodes increments

Data and nodes increments

- Through progressive method, a memory matrix update algorithm is designed to support data increment and adapt to distributed computing needs under dynamic input.

2.3.1. Incremental data and feature expansion

InvF-BLS supports flexible expansion sequences (e.g., data-first, node-first, or mixed iterations). The memory module are updated through the expansion order. Fig. 6 gives a data-first-node-second sequence of feature matrix as an example:

$$\mathbf{A}_0 \xrightarrow{\text{Data } \mathbf{X}_1} \begin{bmatrix}\mathbf{A}_0\\ \mathbf{A}_1\end{bmatrix} \xrightarrow{\text{Node } \mathbf{H}_1} \left[\begin{bmatrix}\mathbf{A}_0\\ \mathbf{A}_1\end{bmatrix}\mathbf{H}_1\right]$$

$$\xrightarrow{\text{Data } \mathbf{X}_2} \left[\begin{bmatrix}\mathbf{A}_0\\ \mathbf{A}_1\end{bmatrix}\mathbf{H}_1\right] \xrightarrow{\text{Node } \mathbf{H}_2} \left[\left[\begin{bmatrix}\mathbf{A}_0\\ \mathbf{A}_1\end{bmatrix}\mathbf{H}_1\right]\mathbf{H}_2\right].$$

Scenario 2: Adding Nodes Changes the Factor Size

Adding new columns enlarges the system

When a new node block $\mathbf{H}_q$ is appended, the feature matrix becomes

$$\mathbf{A}_{k+q} = \begin{bmatrix} \mathbf{A}_k & \mathbf{H}_q \end{bmatrix}.$$

Accordingly, the regularized Gram matrix changes from

$$\mathbf{M}_k = \mathbf{A}_k^{\mathrm{T}} \mathbf{A}_k + \lambda \mathbf{I}$$

to

$$\mathbf{M}_{k+q} = \mathbf{A}_{k+q}^{\mathrm{T}} \mathbf{A}_{k+q} + \lambda \mathbf{I}.$$

The matrix dimension increases from $p \times p$ to $(p+q) \times (p+q)$. Therefore, the old triangular factor $\mathbf{L}_k$ cannot be used directly for the new system.

$$\mathbf{L}_k \mathbf{L}_k^{\mathrm{T}} = \mathbf{M}_k, \qquad \mathbf{L}_{k+q} \mathbf{L}_{k+q}^{\mathrm{T}} = \mathbf{M}_{k+q}.$$

Instead of recomputing a full Cholesky factorization from scratch, we update the factor recursively.

Constructing the New Triangular Factor

Reuse the old factor

Since $\mathbf{L}_k$ is already available, we keep it in the upper-left block of the new factor and introduce two unknown blocks:

$$\mathbf{L}_{k+q} = \left[\begin{array}{c|c} \mathbf{L}_k & \mathbf{0} \\ \hline \mathbf{E} & \mathbf{G} \end{array}\right].$$

Why this block form?

- $\mathbf{L}_k$ reuses the factorization of the old system.
- $\mathbf{E}$ captures the interaction between the old features and the new nodes.
- $\mathbf{G}$ completes the factorization of the newly introduced part.

Goal: solve $\mathbf{E}, \mathbf{G}$ such that $\mathbf{L}_{k+q}\mathbf{L}_{k+q}^{\mathrm{T}} = \mathbf{M}_{k+q}$.

So the update reduces to solving only two new blocks, rather than factoring the whole matrix again.

Fast Update by Block Matching

$$\mathbf{M}_{k+q} = \left[\begin{array}{c|c} \mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k + \lambda\mathbf{I} & \mathbf{A}_k^{\mathrm{T}}\mathbf{H}_q \\ \hline \mathbf{H}_q^{\mathrm{T}}\mathbf{A}_k & \mathbf{H}_q^{\mathrm{T}}\mathbf{H}_q + \lambda\mathbf{I} \end{array}\right] = \mathbf{L}_{k+q}\mathbf{L}_{k+q}^{\mathrm{T}} = \left[\begin{array}{c|c} \mathbf{L}_k\mathbf{L}_k^{\mathrm{T}} & \mathbf{L}_k\mathbf{E}^{\mathrm{T}} \\ \hline \mathbf{E}\mathbf{L}_k^{\mathrm{T}} & \mathbf{E}\mathbf{E}^{\mathrm{T}} + \mathbf{G}\mathbf{G}^{\mathrm{T}} \end{array}\right].$$

Compare the corresponding blocks

Matching the two matrices gives

$$\mathbf{L}_k\mathbf{E}^{\mathrm{T}} = \mathbf{A}_k^{\mathrm{T}}\mathbf{H}_q \quad \Longrightarrow \quad \mathbf{E}^{\mathrm{T}} = \mathcal{FS}\left(\mathbf{L}_k, \mathbf{A}_k^{\mathrm{T}}\mathbf{H}_q\right),$$

and

$$\mathbf{G}\mathbf{G}^{\mathrm{T}} = \left(\mathbf{H}_q^{\mathrm{T}}\mathbf{H}_q + \lambda\mathbf{I}\right) - \mathbf{E}\mathbf{E}^{\mathrm{T}}.$$

The enlarged factor is obtained by

- one triangular solve for $\mathbf{E}$,
- one small Cholesky factorization for $\mathbf{G}$.

Hence, the update is much cheaper than full re-factorization.

Phase 1 (Initialization):

Start by the initiate feature matrix $\mathbf{A}_0$ and labels $\mathbf{Y}_0$, we have $\mathbf{V}_0 = \mathbf{A}_0^{\mathrm{T}}\mathbf{Y}_0$. Setting $\mathbf{L}_0 = \mathbf{R}_0^{\mathrm{T}}$ and applying Cholesky decomposition on

$$\mathbf{M}_0 = \mathbf{A}_0^{\mathrm{T}}\mathbf{A}_0 + \lambda\mathbf{I} = \mathbf{L}_0\mathbf{L}_0^{\mathrm{T}} \triangleq \mathbf{R}_0^{\mathrm{T}}\mathbf{R}_0.$$

Therefore, $\mathbf{W}_0$ gets from the initial memory module $(\mathbf{R}_0, \mathbf{V}_0)$:

$$\mathbf{L}_0\mathbf{L}_0^{\mathrm{T}}\mathbf{W}_0 = \mathbf{R}_0^{\mathrm{T}}\mathbf{R}_0\mathbf{W}_0 = \mathbf{V}_0.$$

Phase 2 (Data Incremental Update):

For new data block $(\mathbf{A}_k, \mathbf{Y}_k)$, the memory module $(\mathbf{R}_k, \mathbf{V}_k)$ is updated as (7) and (8):

$$\begin{cases} \begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix} = \mathbf{Q}_k\mathbf{R}_k, \\ \mathbf{V}_k = \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k. \end{cases} \tag{10}$$

Moreover, we can conclude

$$\mathbf{L}_k = \mathbf{R}_k^{\mathrm{T}},\ 1 \le k \le N, \tag{11}$$

where $\mathbf{M}_k = \mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k + \lambda\mathbf{I} = \mathbf{L}_k\mathbf{L}_k^{\mathrm{T}}$, proved in Theorem 3.2 of Section 3. It is a bridge connecting data and node increments.

Phase 3 (Node Incremental Update):

When adding nodes $\mathbf{H}_k$, based on the bright $\mathbf{L}_k = \mathbf{R}_k^{\mathrm{T}}$ and node incremental methods in Chen et al.[34], $\mathbf{L}_k^*$ could be updated by :

$$\mathbf{L}_k^* = \begin{bmatrix} \mathbf{L}_k & 0 \\ \mathbf{E} & \mathbf{G} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_k^{\mathrm{T}} & 0 \\ \mathbf{E} & \mathbf{G} \end{bmatrix} \triangleq \left(\mathbf{R}_k^*\right)^{\mathrm{T}}, \tag{12}$$

where

$$\begin{cases} \mathbf{E}^{\mathrm{T}} = \mathcal{BS}\left(\mathbf{L}_k, \mathbf{A}_{0:k}^{\mathrm{T}}\mathbf{H}_k\right), \\ \mathbf{G}\mathbf{G}^{\mathrm{T}} = \mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k - \mathbf{E}\mathbf{E}^{\mathrm{T}} + \lambda\mathbf{I}. \end{cases}$$

And $\mathbf{V}_k^*$ can be computed from $\mathbf{V}_k$:

$$\mathbf{V}_k^* = \left[\mathbf{A}_{0:k}, \mathbf{H}_k\right]^{\mathrm{T}}\mathbf{Y}_{0:k} = \begin{bmatrix} \mathbf{V}_k \\ \mathbf{H}_k^{\mathrm{T}}\mathbf{Y}_{0:k} \end{bmatrix}. \tag{13}$$

Summary：Efficient and high-precision nodes increments
(Accuracy approximation)

$$\mathbf{W}_k = \mathcal{BS}\left(\left(\mathbf{L}_k^*\right)^{\mathrm{T}}, \mathcal{FS}\left(\mathbf{L}_k^*, \mathbf{V}_k^*\right) \right),$$

where:

$$\begin{cases} \mathbf{L}_k^* = \begin{bmatrix} \mathcal{QR}\left(\begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix} \right)^{\mathrm{T}} & 0 \\ \mathbf{E} & \mathbf{G} \end{bmatrix}, \\ \mathbf{V}_k^* = \begin{bmatrix} \mathbf{R}_{k-1}^{\mathrm{T}} \mathbf{R}_{k-1} \mathbf{W}_{k-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{Y}_k \\ \mathbf{H}_k^{\mathrm{T}} \mathbf{Y}_{0:k} \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}} \mathbf{Y}_k \\ \mathbf{H}_k^{\mathrm{T}} \mathbf{Y}_{0:k} \end{bmatrix}. \end{cases}$$

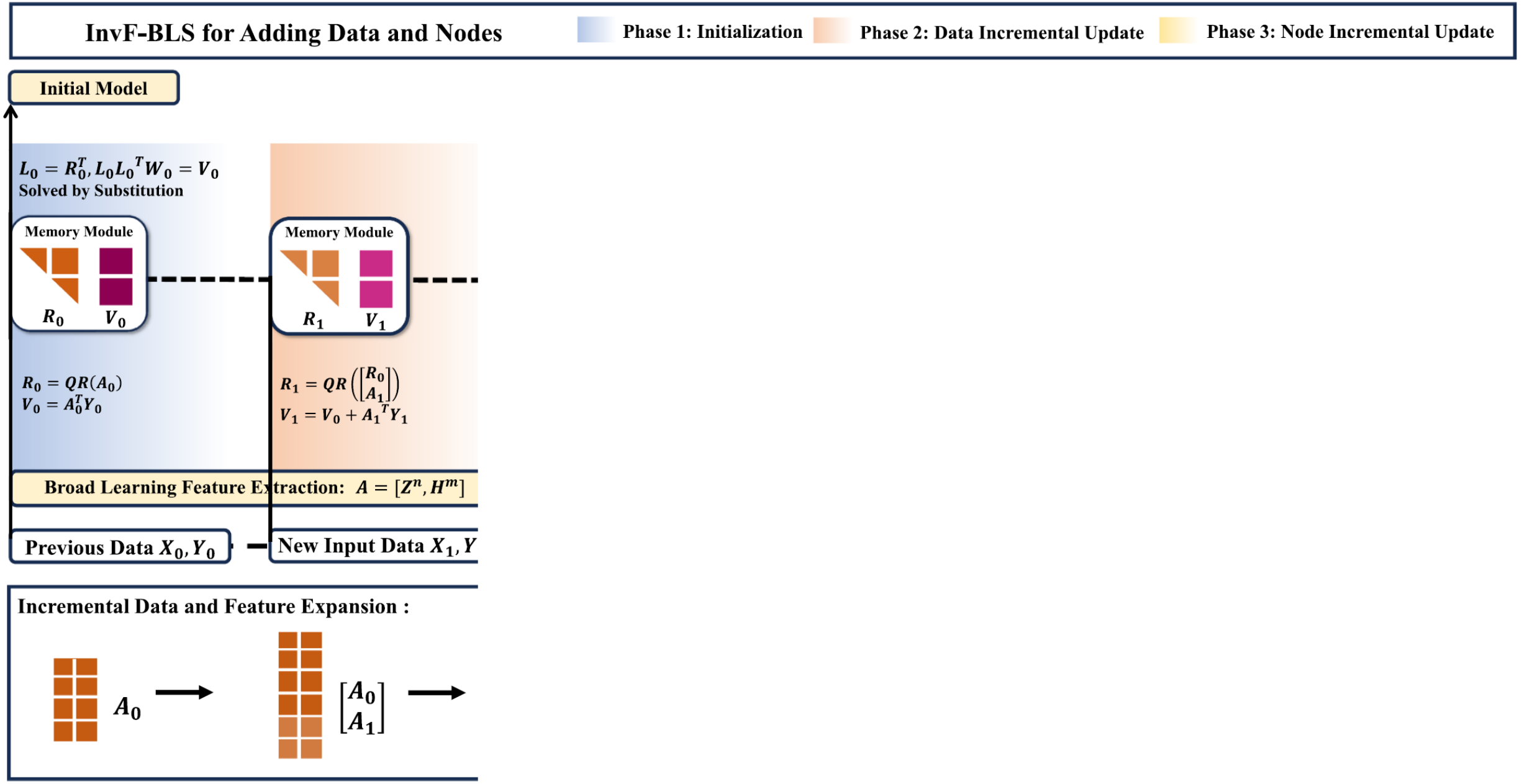
The recursive update process follows a chain:

$$\mathbf{R}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (10)}} \mathbf{R}_k \xrightarrow[\text{Bridge}]{\text{Transposition in (11)}} \mathbf{L}_k \xrightarrow[\text{Node Incremental}]{\text{Update in (12)}} \mathbf{L}_k^*$$

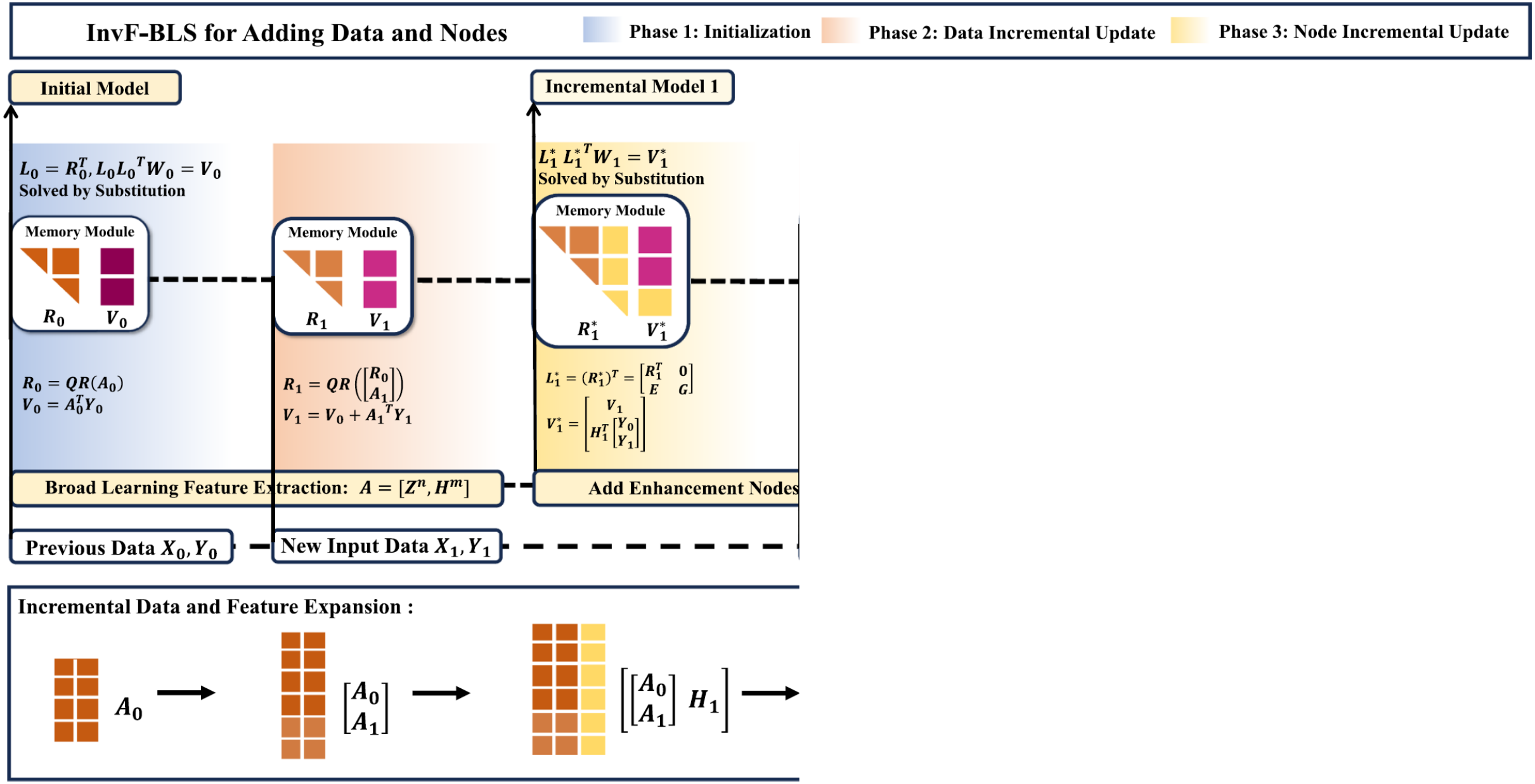
$$\text{and } \mathbf{V}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (10)}} \mathbf{V}_k \xrightarrow[\text{Node Incremental}]{\text{Update in (13)}} \mathbf{V}_k^*$$

$$\Rightarrow \mathbf{W}_{k-1} \xrightarrow{\text{Update in (14)}} \mathbf{W}_k.$$

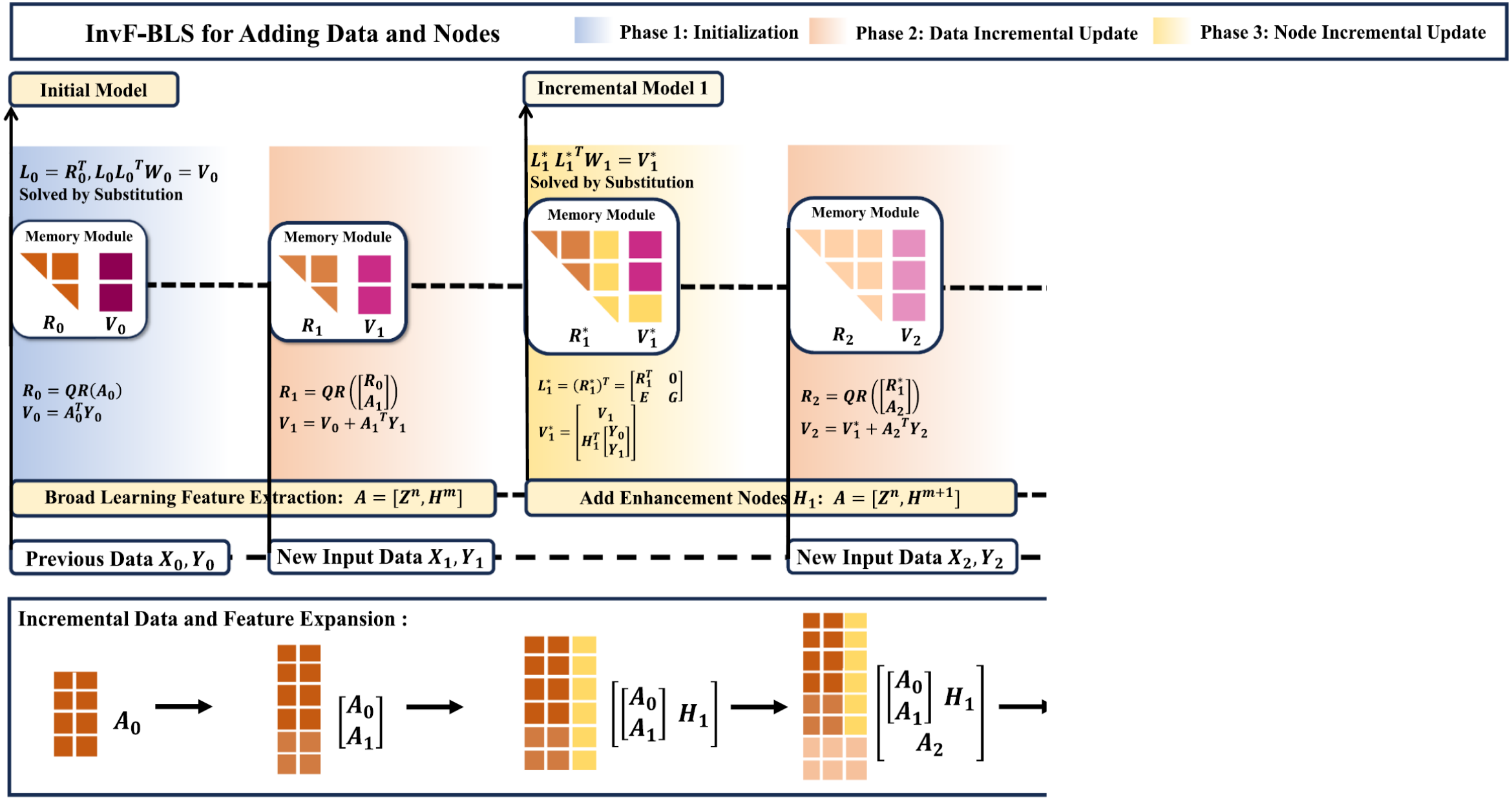
Efficient incremental calculations in parallelism：Real-time data flow+nodes incremental c



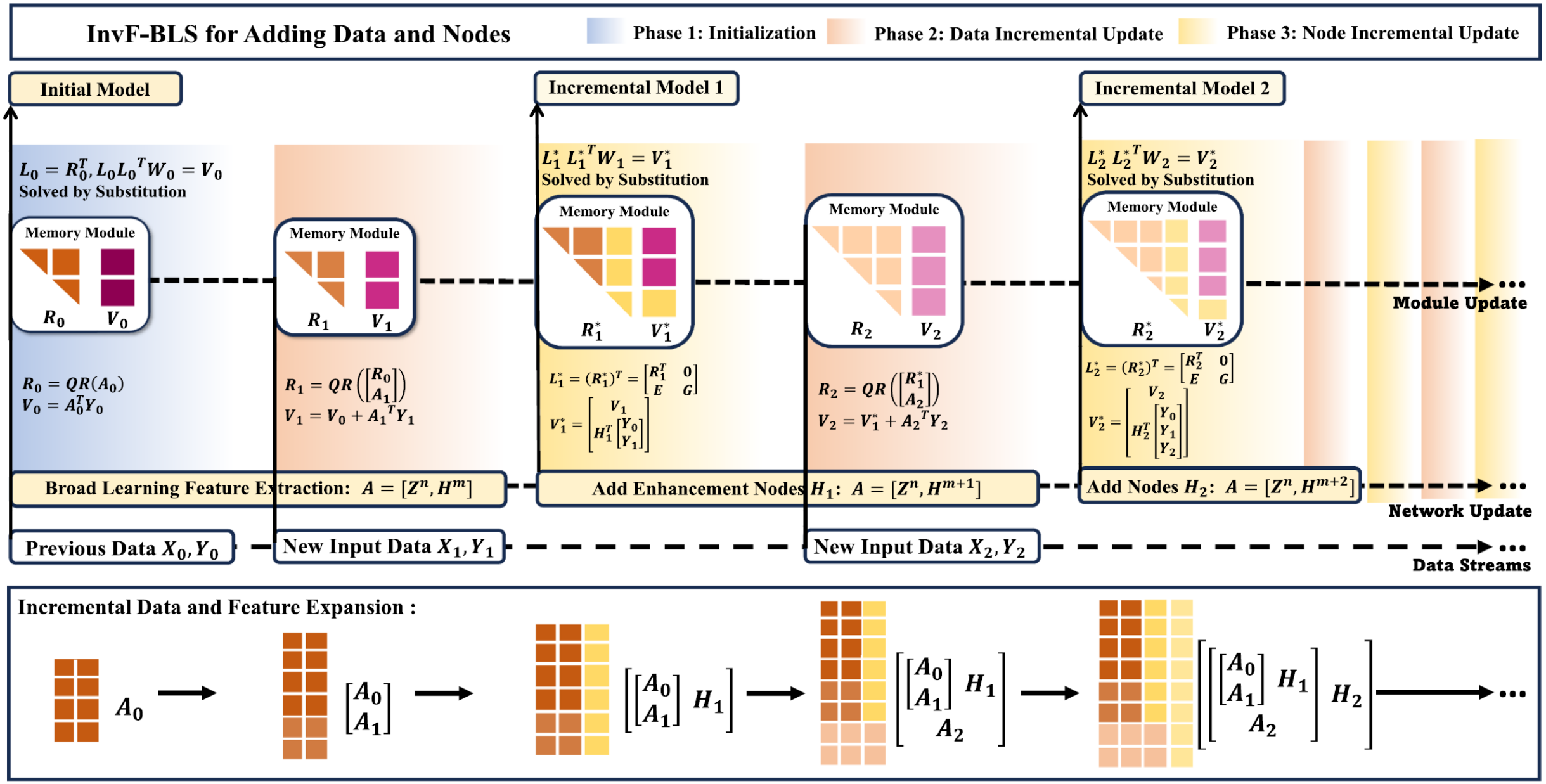
Real-time data flow+nodes incremental calculation



Real-time data flow+nodes+data incremental calculation



Real-time data flow+nodes+data+nodes incremental



Efficient and high-precision data incremental calculation

$$\mathbf{W}_k = \mathcal{BS}\left(\mathbf{R}_k, \mathcal{FS}\left(\mathbf{R}_k^{\mathrm{T}}, \mathbf{V}_k\right)\right),$$

$$\begin{cases} \mathbf{R}_k = \mathcal{QR}\left(\begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix}\right), \\ \mathbf{V}_k = \mathbf{R}_{k-1}^{\mathrm{T}}\mathbf{R}_{k-1}\mathbf{W}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k = \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k. \end{cases}$$

This process enables efficient adaptation to streaming da
recursive update chain:

$$\mathbf{R}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (7)}} \mathbf{R}_k \quad \text{and} \quad \mathbf{V}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in(8)}} \mathbf{V}_k$$

$$\Rightarrow \mathbf{W}_{k-1} \xrightarrow{\text{Update in(9)}} \mathbf{W}_k.$$

Efficient and high-precision node increments (precision approximation)

$$\mathbf{W}_k = \mathcal{BS}\left(\left(\mathbf{L}_k^*\right)^{\mathrm{T}}, \mathcal{FS}\left(\mathbf{L}_k^*, \mathbf{V}_k^*\right)\right),$$

where:

$$\begin{cases} \mathbf{L}_k^* = \begin{bmatrix} \mathcal{QR}\left(\begin{bmatrix} \mathbf{R}_{k-1} \\ \mathbf{A}_k \end{bmatrix}\right)^{\mathrm{T}} & 0 \\ \mathbf{E} & \mathbf{G} \end{bmatrix}, \\ \mathbf{V}_k^* = \begin{bmatrix} \mathbf{R}_{k-1}^{\mathrm{T}}\mathbf{R}_{k-1}\mathbf{W}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k \\ \mathbf{H}_k^{\mathrm{T}}\mathbf{Y}_{0:k} \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{k-1} + \mathbf{A}_k^{\mathrm{T}}\mathbf{Y}_k \\ \mathbf{H}_k^{\mathrm{T}}\mathbf{Y}_{0:k} \end{bmatrix}. \end{cases}$$

The recursive update process follows a chain:

$$\mathbf{R}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (10)}} \mathbf{R}_k \xrightarrow[\text{Bridge}]{\text{Transposition in (11)}} \mathbf{L}_k \xrightarrow[\text{Node Incremental}]{\text{Update in (12)}} \mathbf{L}_k^*$$

$$\text{and } \mathbf{V}_{k-1} \xrightarrow[\text{Data Incremental}]{\text{Update in (10)}} \mathbf{V}_k \xrightarrow[\text{Node Incremental}]{\text{Update in (13)}} \mathbf{V}_k^*$$

$$\Rightarrow \mathbf{W}_{k-1} \xrightarrow{\text{Update in (14)}} \mathbf{W}_k.$$

Time Efficiency and Accuracy

Table 5

Results of final testing accuracy and total train time on real-world datasets.

Dataset	Incremental BLS		TiBLS		Approximation Method		RI-BLS		InvF-BLS	
	Acc (%)	Time (s)	Acc (%)	Time (s)	Acc (%)	Time (s)	Acc (%)	Time (s)	Acc (%)	Time (s)
LED	74.05	1838.14	74.05	108.21	74.03	61.93	74.04	28.28	**74.08**	**4.76**
CIFAR10	88.59	224.92	88.60	46.98	24.42	24.82	88.59	14.67	**88.64**	**1.71**
CIFAR100	64.92	87.58	64.75	18.60	52.46	8.86	64.92	6.25	**65.28**	**0.76**
Waveform	82.50	0.31	82.63	0.59	81.25	0.18	82.50	0.17	**82.80**	**0.01**
Letter	92.72	8.40	93.40	6.83	91.22	3.66	92.72	2.89	**95.30**	**0.26**
Pendigits	98.03	1.01	97.76	1.27	98.02	0.47	98.02	0.38	**98.90**	**0.02**
Shuttle	98.61	86.57	98.52	24.07	98.61	6.97	98.61	5.56	**99.03**	**0.64**

Time Efficiency and Accuracy

Table 8

Accuracy and training time with different numbers of added enhancement nodes on MNIST.

M_1, M_2	N_1, N_2, N_3	One-shot Accuracy(%)	One-shot Training time(s)	Incremental BLS Accuracy(%)	Incremental BLS Training time(s)	InvF-BLS Accuracy(%)	InvF-BLS Training time(s)
10000, 100	10, 10, 2000	97.73	6.963	95.86	3.469	**97.82**	**2.083**
10000, 500	10, 10, 2000	97.90	17.340	96.62	8.586	**98.32**	**5.386**
10000, 1000	10, 10, 2000	97.99	37.407	97.04	19.656	**98.69**	**13.016**
10000, 1500	10, 10, 2000	97.93	66.411	97.17	34.420	**98.77**	**24.501**
10000, 2000	10, 10, 2000	98.28	104.472	97.36	53.863	**98.89**	**40.064**
10000, 100	20, 10, 2000	97.83	7.754	96.12	3.735	**97.87**	**2.206**
10000, 500	20, 10, 2000	97.93	18.509	96.98	9.054	**98.49**	**5.655**
10000, 1000	20, 10, 2000	98.30	39.075	97.28	20.192	**98.70**	**13.616**
10000, 1500	20, 10, 2000	98.45	68.173	97.38	35.242	**98.91**	**24.667**
10000, 2000	20, 10, 2000	98.07	106.888	97.50	54.417	**98.89**	**40.710**

Space and Memory Efficiency

Table 2

Comparison of Number of Operations, Time Complexity, Space Requirements and Space Complexity. Note: N^* is the number of new samples; N is the number of original samples; p is the total number of nodes; c is the number of classes.

Method	Number of Operations (Flops)	Time Complexity	Space Requirements	Space Complexity
InvF-BLS (Ours)	*Steps:* - Incremental R-Factor Update Eq. (7): N^*p^2 - Right-Hand Side Update Eq. (8): $2N^*pc$ - Weight Calculation Eq. (9): $2p^2c$ **Total: $N^*p^2 + 2N^*pc + 2p^2c$**	$O(N^*p^2 + N^*pc + p^2c)$	$\frac{1}{2}p^2 + \frac{1}{2}p + 2pc$	$O(p^2 + pc)$
RI-BLS [22]	*Steps (from* [22]*):* - Memory Matrix U* Update (Eq. 11): $2N^*p^2$ - Memory Matrix V* Update (Eq. 12): $2N^*pc$ - Weight Calculation (Eq. 15): $p^3 + 2p^2c$ **Total: $p^3 + 2N^*p^2 + 2N^*pc + 2p^2c$**	$O(p^3 + N^*p^2 + N^*pc + p^2c)$	$3p^2 + 3pc$	$O(p^2 + pc)$
Approximation Method [46]	$p^3 + 4N^*p^2 + 2Npc + 2N^*pc$	$O(p^3 + N^*p^2 + Npc + N^*pc)$	$2p^2 + 4N^*p + 2Np + N^*p$ $+Np + Nc + N^*c + pc$	$O(p^2 + Np + N^*p + Nc + N^*c + pc)$
Incremental BLS [14]	$p^3 + 6NN^*p + 4N^*p^2 + 4N^*pc$ or $N^{*3} + 8NN^*p + 2pN^{*2} + 4N^*pc$	$O(p^3 + NN^*p + N^*p^2 + N^*pc)$ or $O(N^{*3} + NN^*p + pN^{*2} + N^*pc)$	$3N^*p + 2Np + NN^* + N^*c + pc$ or $2N^{*2} + 5Np + NN^* + N^*c + pc$	$O(N^*p + Np + NN^* + N^*c + pc)$ or $O(N^{*2} + N^*p + Np + NN^* + N^*c + pc)$

Memory and data volume not depend on N, , It only relates to neuron nodes, p

Lots of *N* in other approaches

Real-time computing of edge embodied intelligence

Offline intelligence

via:

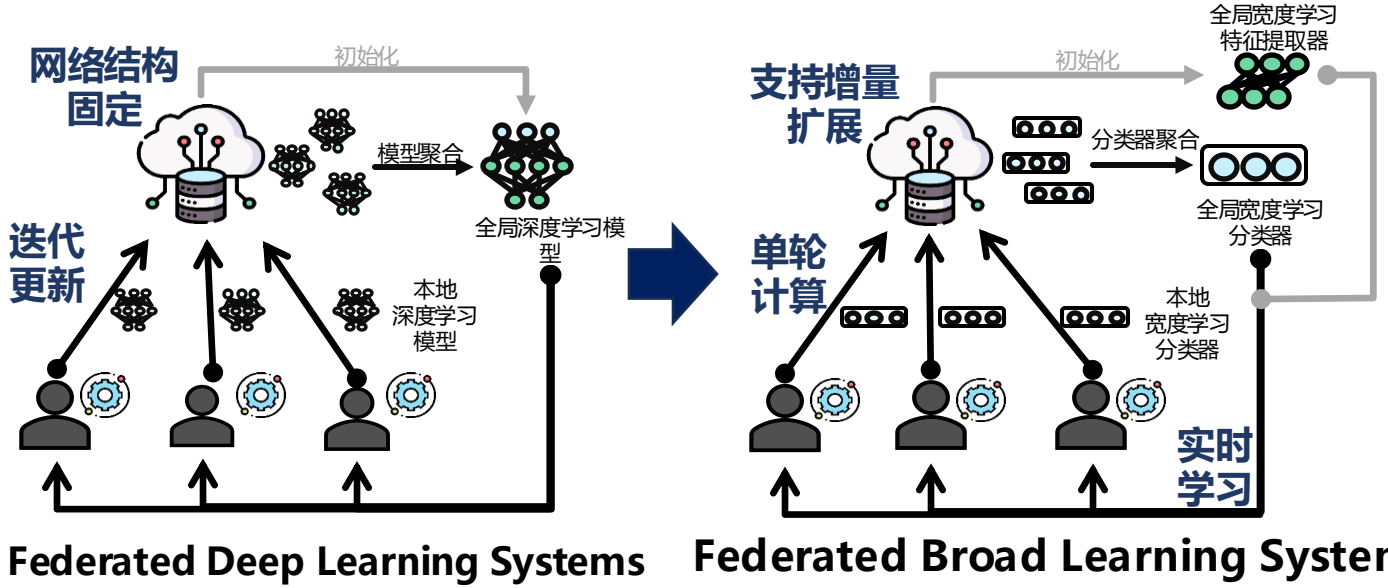
BLS provides Real-time computing in a dynamic open environment

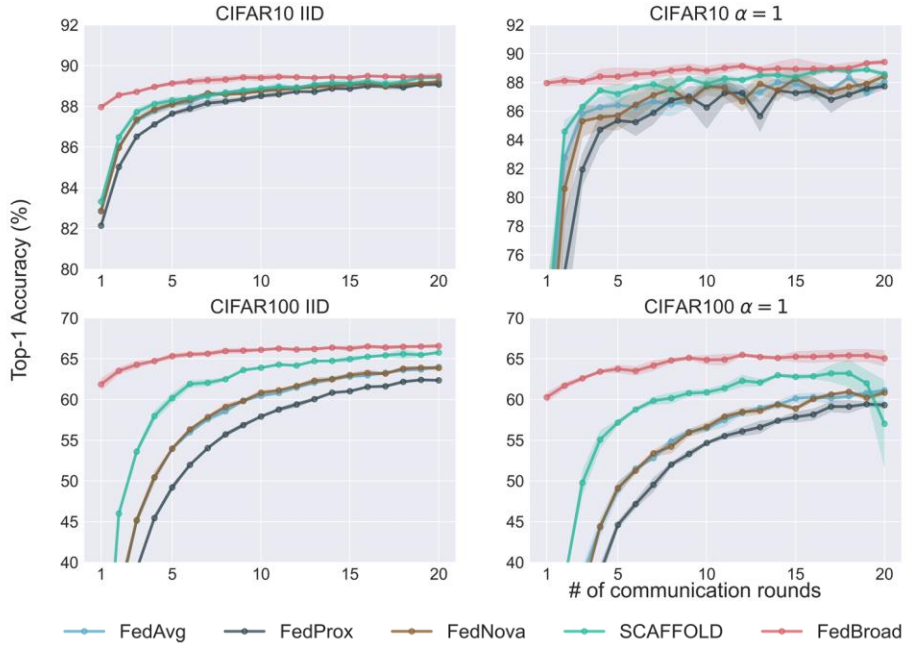
Broad federated learning – a federated modeling approach aimed at low communication overhead

effectively solving the problem of excessive communication overhead in traditional federated learning methods and providing a new paradigm for secure distributed deployment of lightweight neural networks in open environments.

A broad federated learning system that communicates efficiently

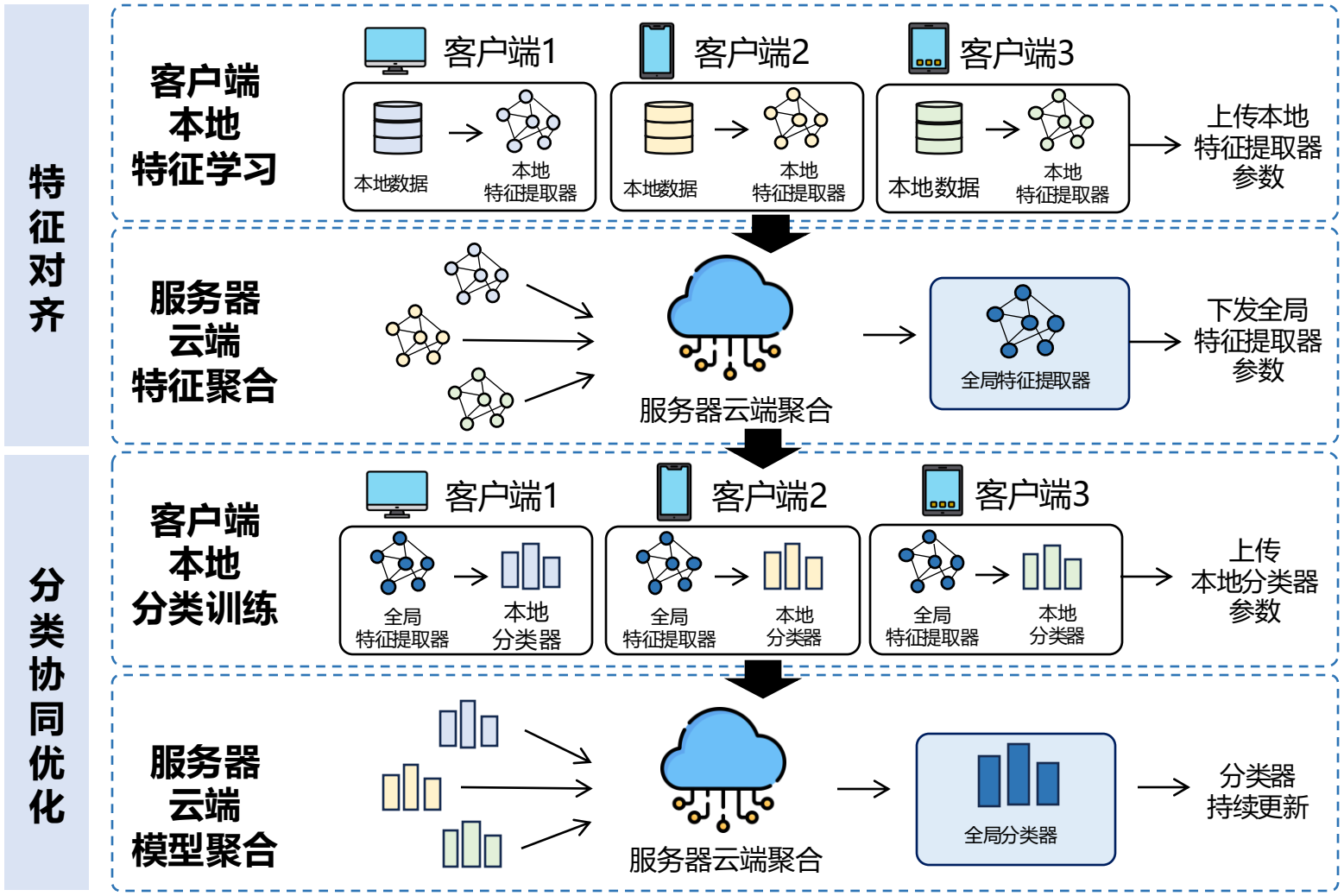
✓ Leveraging the advantages BLS, feature alignment and near-end regex mechanisms are introduced to enhance the generalization capability of federated systems under heterogeneous data, achieving efficient federated learning for communication.





在保证模型精度的前提下显著减少通信轮数

Broad federated learning – a federated modeling approach aimed at low communication overhead



Core challenges of traditional federated learning

In an open and complex environment, cloud-edge-end intelligence requires a privacy-protection collaborative modeling method that offers lower latency, higher efficiency, and sustainable updates.

Broad federal learning

By utilizing the **single-round analytical training** feature of breadth learning, it replaces traditional federated learning with multi-turn iterative optimization to achieve efficient collaborative modeling。

Data privacy protection
Data never leaves the local area

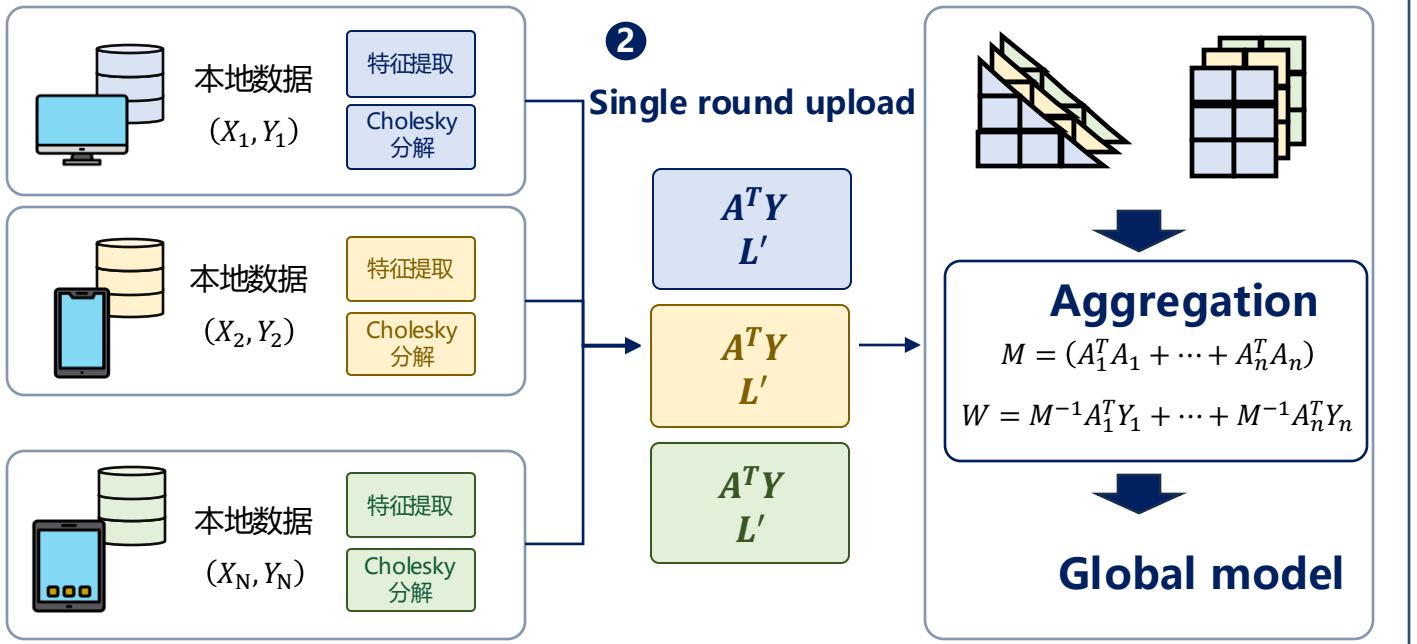
Reduce multiple rounds of iteration

Low communication overhead
Only transmit model parameters, averaging fewer communication turns22.9×倍

Supports continuous learning
Adapt to a dynamically changing environment

Invert-free Broad Federated Learning

Methodological framework



Core advantages

Single-round collaborative communication
一次交互完成建模，提升实时协同能力。

Heterogeneous、Distributed、Robust
结果相对数据分布不变，适应复杂场景。

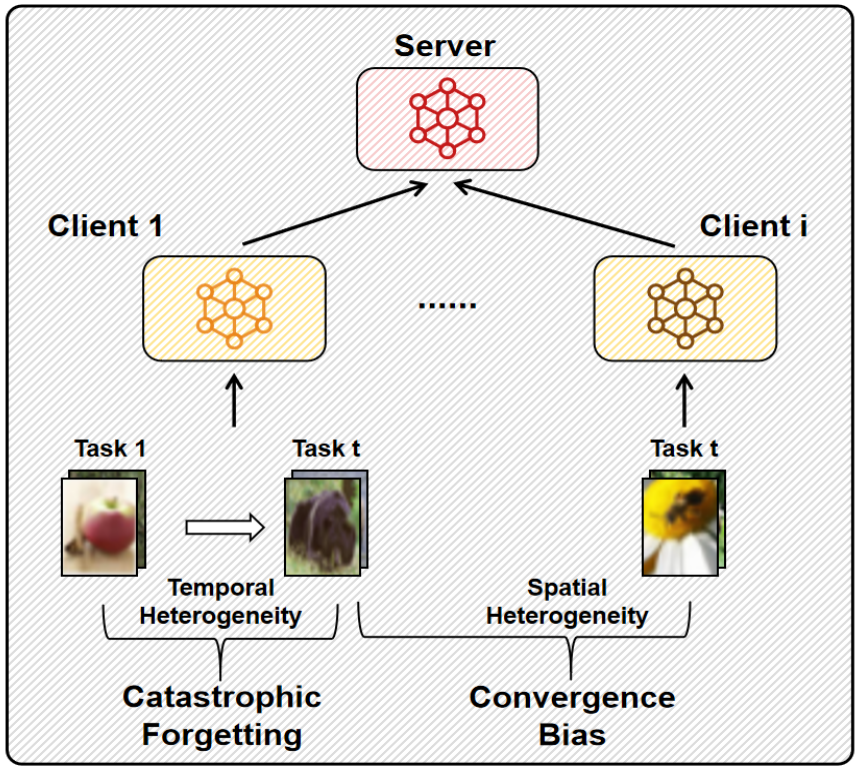
Efficient analytical solutions
避免显式求逆，降低边缘端资源占用。

Stable online updates
减弱病态矩阵影响，提升更新稳定性。

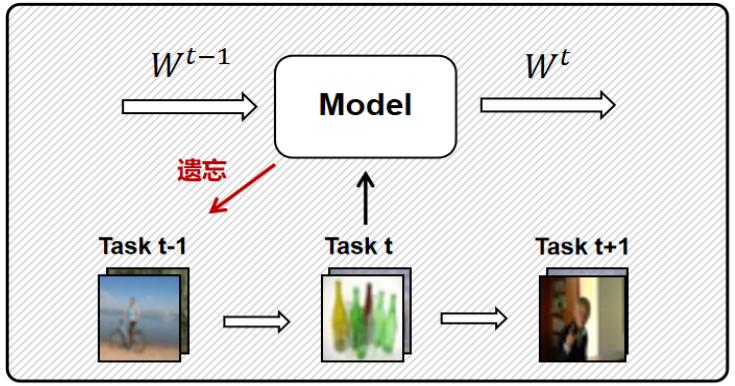
Inverse-freefederated learning combines the advantages of single-wheel, efficient, robust, and stable.This further enhances the application capability of broad federated learning in complex scenarios.

A federated learning framework for continuous data flows

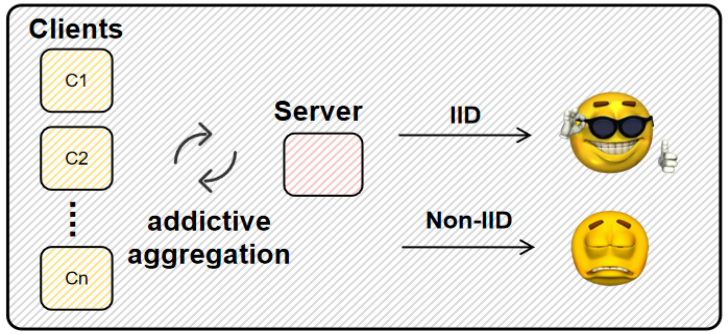
Traditional federated learning usually assumes fixed tasks and struggles to cope with knowledge forgetting when new tasks arrive, thus introducing the concept of continual learning. Federated continual learning achieves continuous adaptation of models to new knowledge under privacy protection and distributed constraints, while striving to retain old knowledge。



(a) 联邦持续学习的两大挑战



(b) 灾难性遗忘



(c) 不一致收敛

难点：数据在时间和空间上的分布都不均衡，数据时间异质性会引发模型对早期知识的灾难性遗忘；而数据空间异质性会导致聚合损失曲面发散，引发不一致收敛。在多轮学习中，两者还会存在叠加效应，进一步加剧模型性能的退化。

66

2024 年广州市科普作品大赛获奖作品榜单

2024 年广州市优秀科普作品（图书）

序号	图书名称	主要作者	出版社	推荐单位（推荐人）
1	《“口袋里的人工智能”系列丛书》	陈俊龙（总主编）	广东科技出版社	广东科技出版社
2	《癌症那些事》	万德森	广东科技出版社	广东科技出版社
3	《藏在节气里的中医》	曾科学	中国中医药出版社	广东省第二中医院
4	《常见心血管病调养功法》	彭锐（广州中医药大学第一附属医院）	广东科技出版社	自荐（彭锐）
5	《常用补益中药鉴别与应用》	夏鑫华	广东科技出版社	广州医科大学附属第一医院
6	《读山、听史、观自然白云山科普丛书》	广州市白云山风景名胜区管理局	花城出版社	广州市白云山风景名胜区管理局、广东花城

Currently, it has exported Mongolian, German, Arabic, Malay, and other foreign language copyrights

Artificial intelligence technology empowers the development of digital economy industry

Thank you, everyone!

中国科协年会
Annual Meeting of the China Association for Science and Technology

华南理工大学
South China University of Technology