

Embedding domain knowledge: Inductive biases and algorithmic alignment in Machine Learning

Serio Agriesti; Francisco Camara Pereira
samaa@dtu.dk

DTU – Department of technology, management and economics

The context

The APEX project

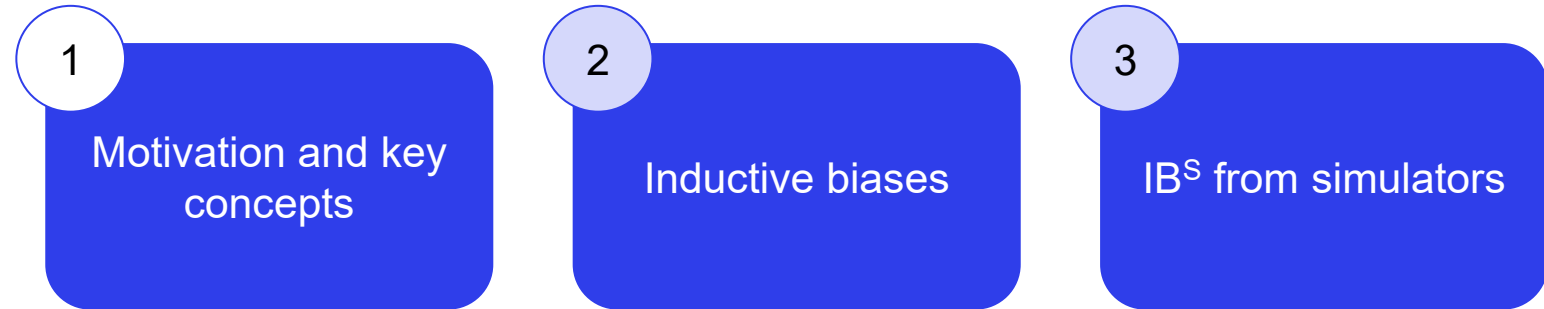
Artificial intelligence for
policy excellence in the
climate crisis

Objective 1

Accelerating large-scale simulators
through ML

Objective 2

Ensuring that the adopted ML
solutions are **robust out-of-
distribution**



Outline



Aim of the presentation

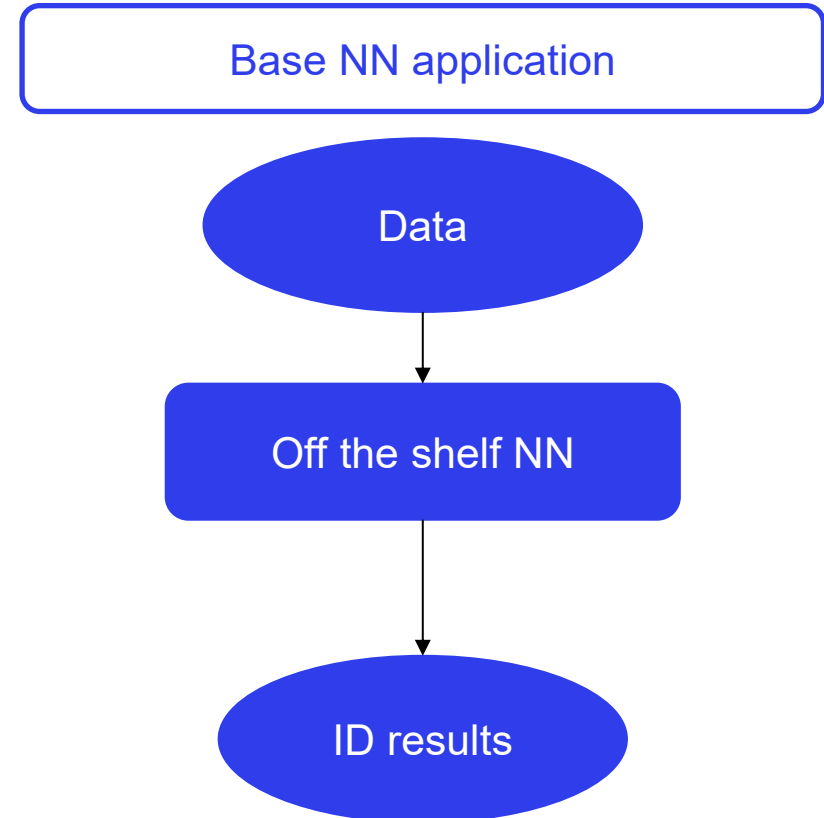
Our work tries to lay the theoretical background:

1. To expand concepts from **algorithmic alignment** to the **ensembles of algorithms that are simulators.**
2. To identify the portions of **domain knowledge from simulators** that can be formulated as **inductive biases**

Motivation

The subject

Off the shelf NN (neural network) approaches are effective in distribution (ID) but **struggle out of distribution** (OOD)



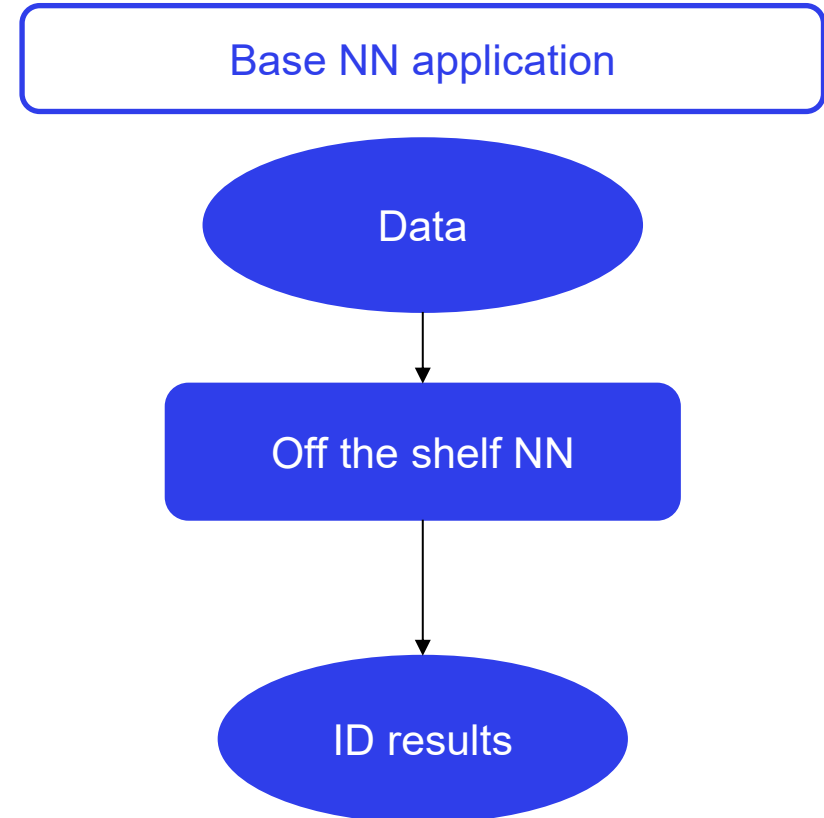
Motivation

The subject

Off the shelf NN (neural network) approaches are effective in distribution (ID) but **struggle out of distribution** (OOD)

The problem

NN cannot be **reliably** used for **scenario evaluation** in most cases that involve a big **disruption** (or can leverage **few data**)



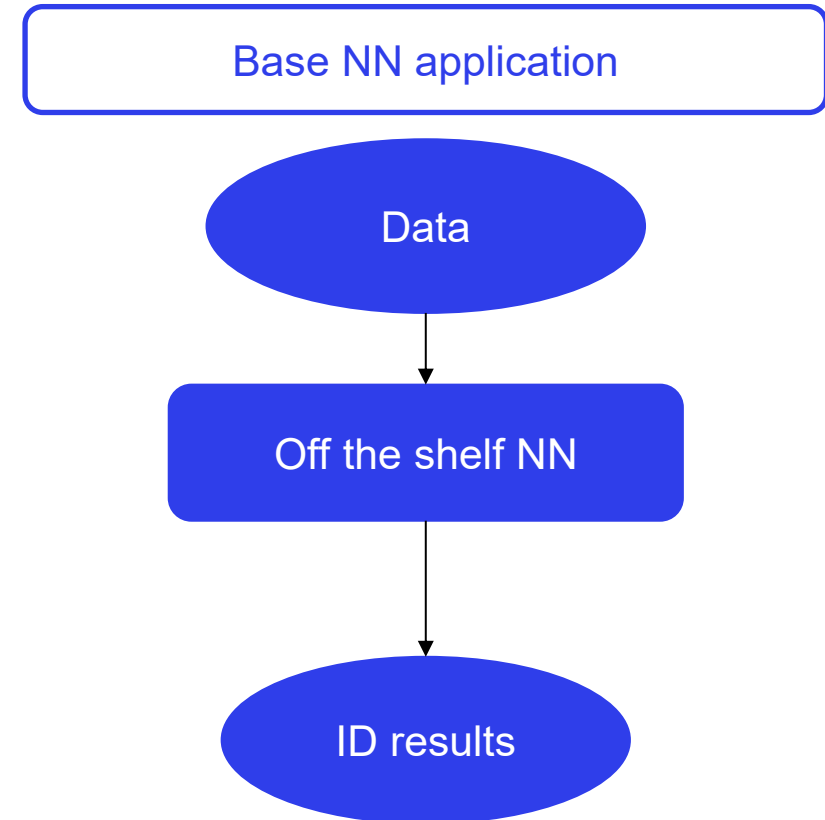
Motivation

The subject

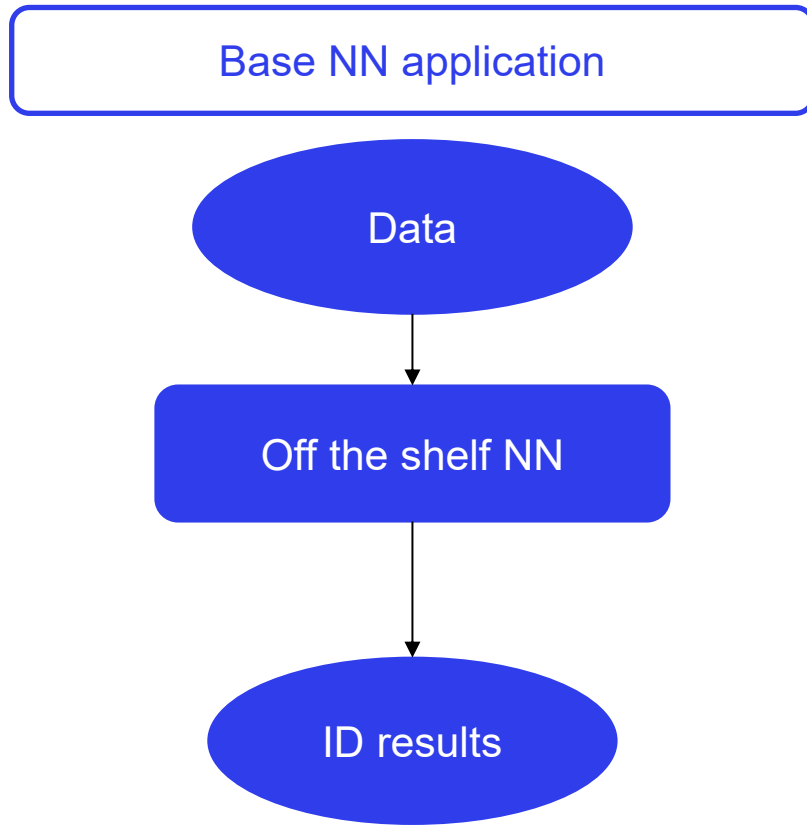
Off the shelf NN (neural network) approaches are effective in distribution (ID) but **struggle out of distribution** (OOD)

The problem

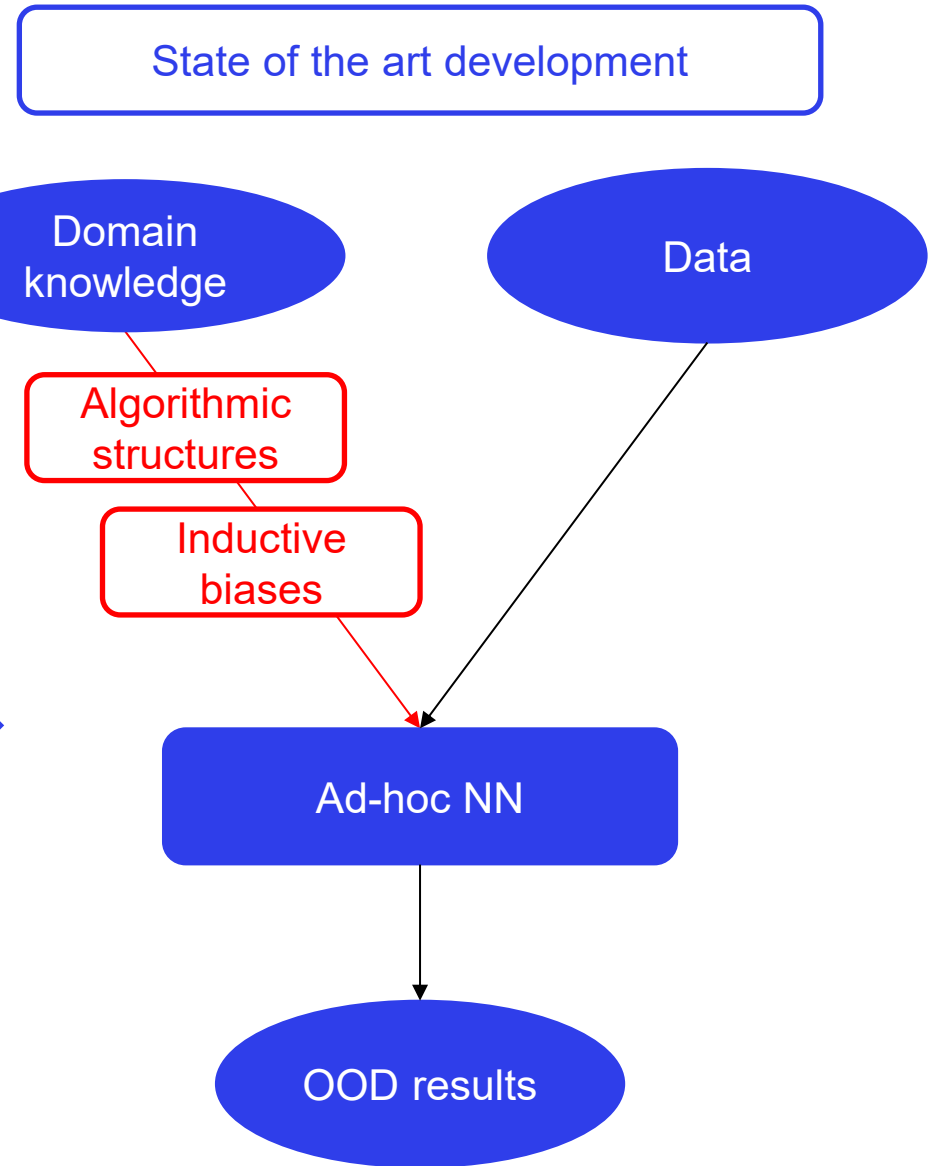
NN cannot be **reliably** used for **scenario evaluation** in most cases that involve a big **disruption** (or can leverage **few data**)



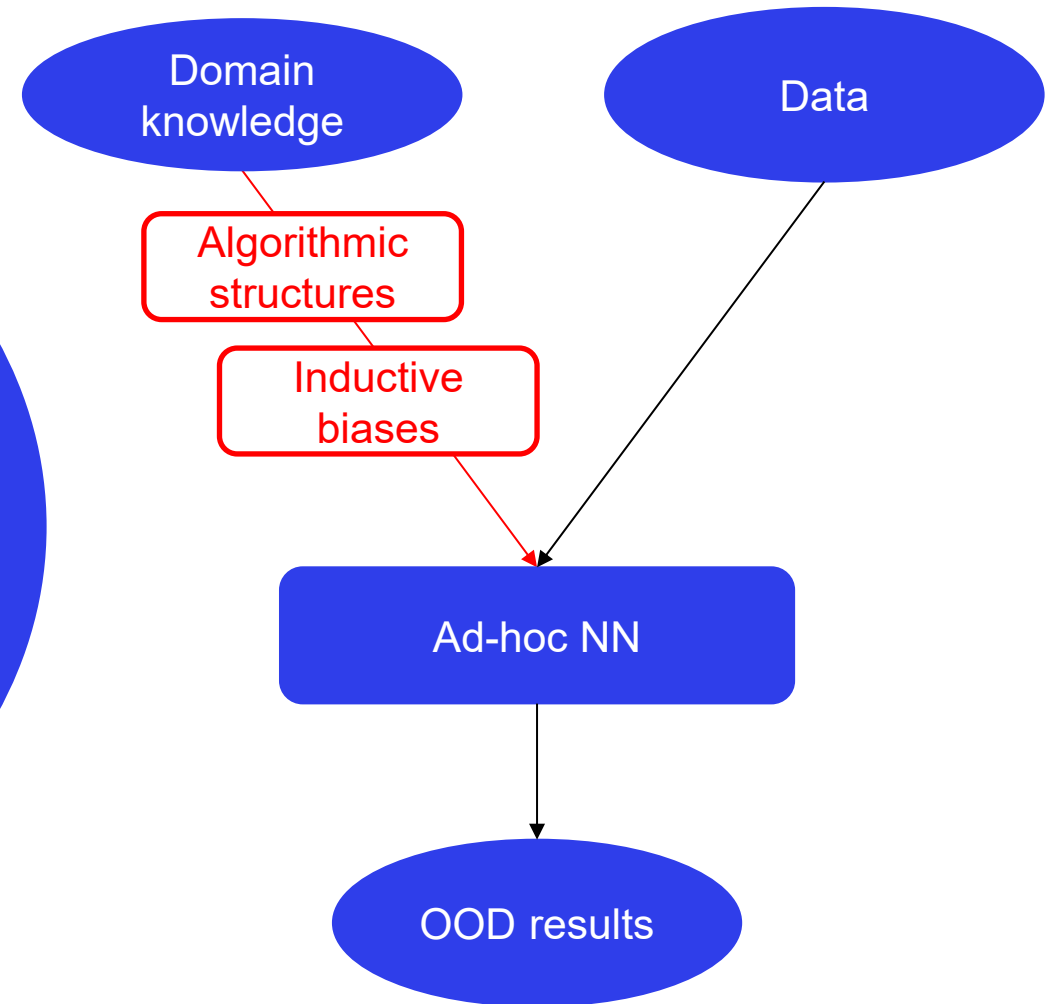
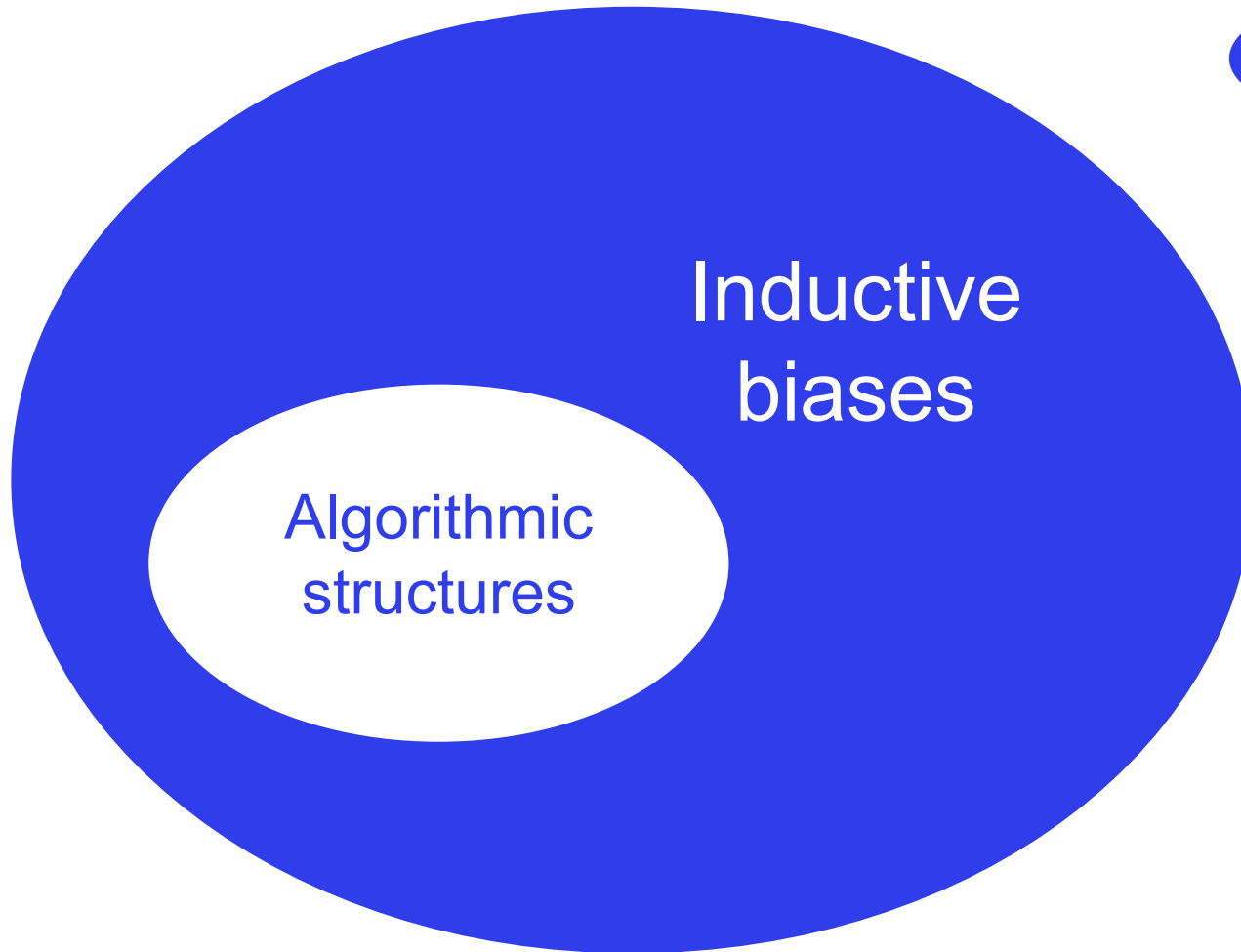
State of the art



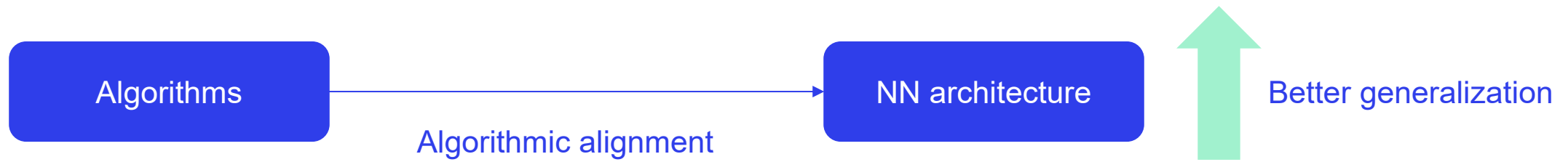
Trend



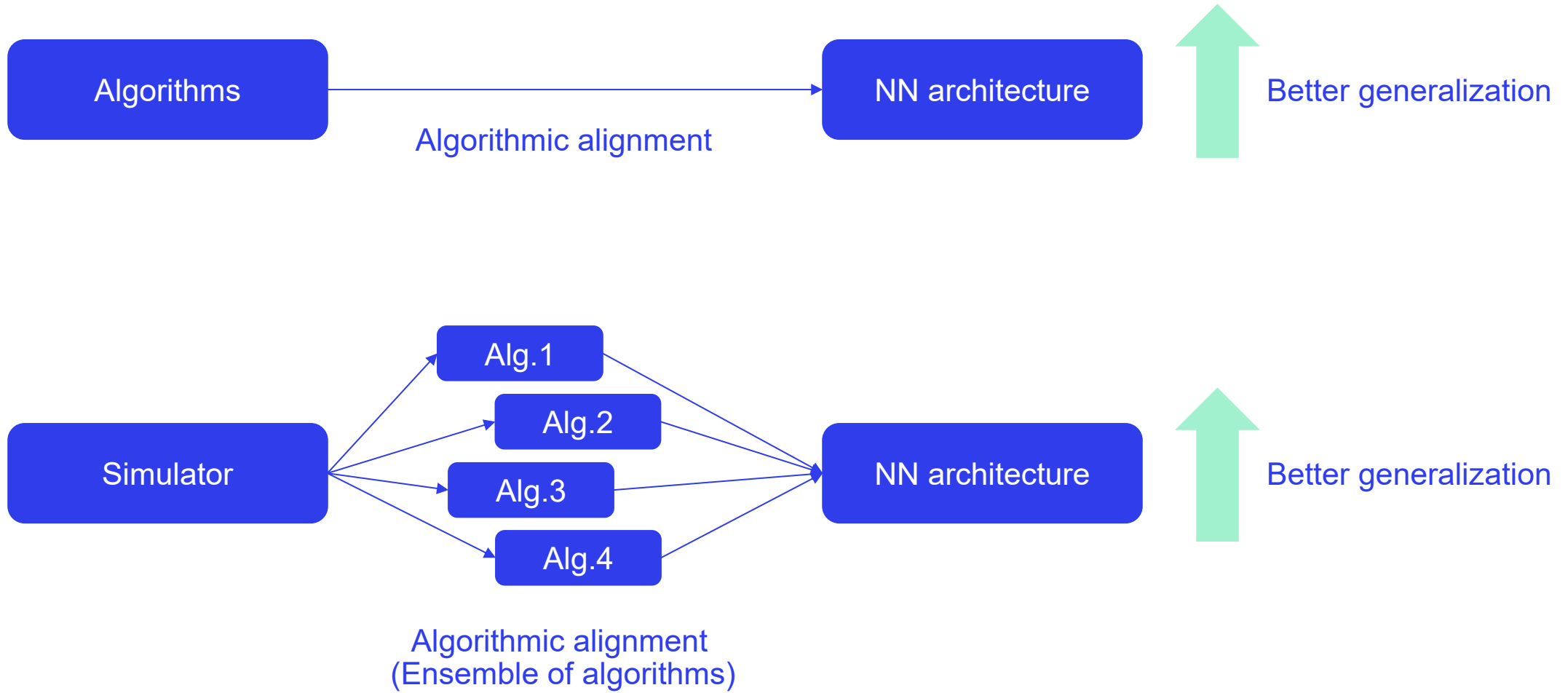
IB definition



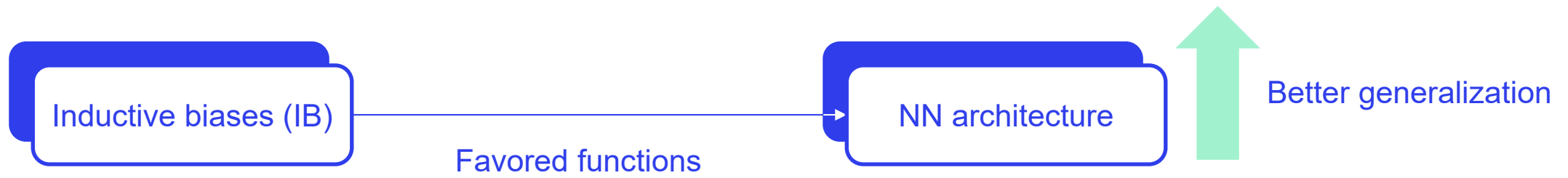
Motivation



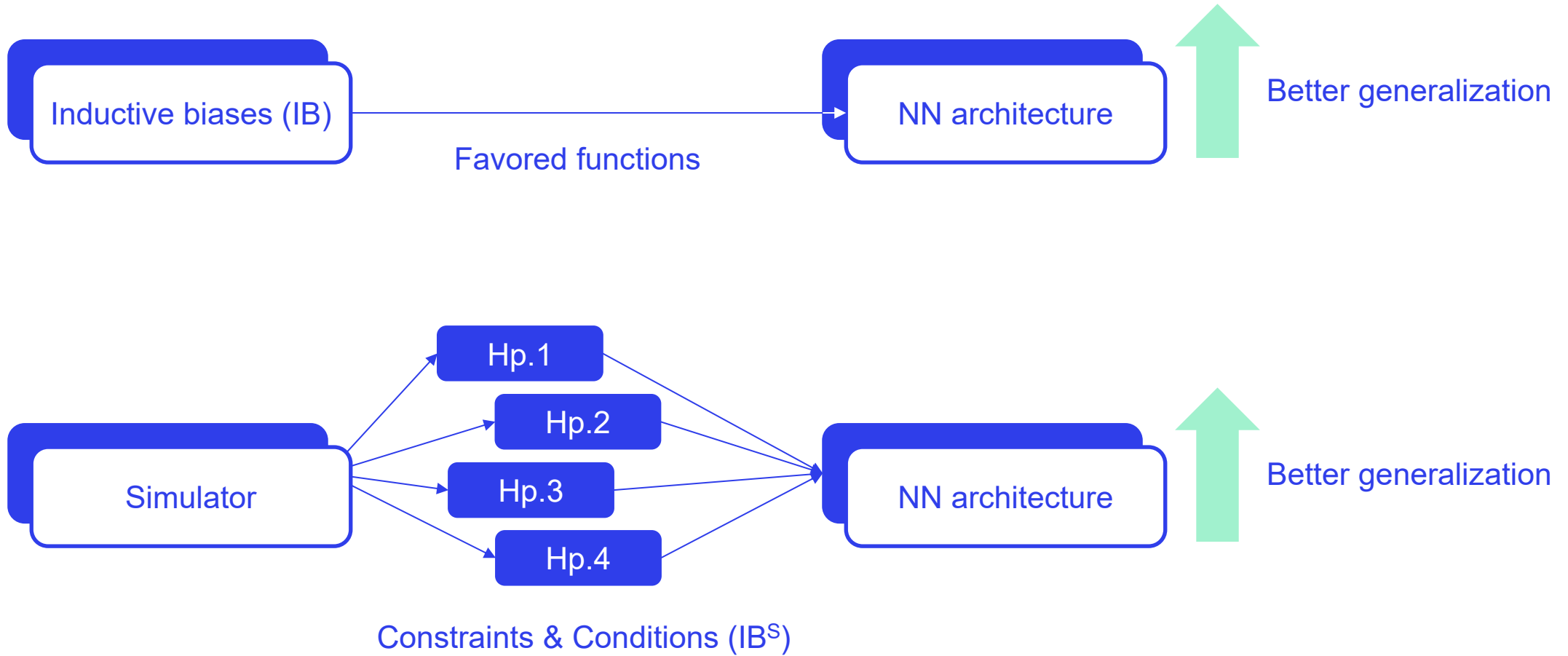
Motivation



Motivation



Motivation



Motivation

NN

- Fast
- Exploit hidden representations
- Identify patterns



-
- Struggle OOD
 - Less suited for scenario analysis



Simulators

- Consistent behaviour
- Directly embeds major dynamics



-
- Slow
 - Algorithmic bottleneck¹



¹ The richness of information from the real world is lost due to which and how data is recorded



Outline



Inductive biases

Definition 1

An inductive bias is a **preference towards a certain type of generalization** over others. Multiple functions may fit the training data, the IB should ideally favour the ones that perform better OOD [Wilson and Frank, 2023]

Inductive biases

Definition 1

An inductive bias is a **preference towards a certain type of generalization** over others. Multiple functions may fit the training data, the IB should ideally favour the ones that perform better OOD [Wilson and Frank, 2023]

Definition 2

An inductive bias frames **which continuation (OOD) is simpler** to learn for the learner. A learner is biased towards C_1 over C_2 if the training set and its extension according to C_1 has a **shorter description length** compared to C_2 [Kharitonov and Chaabouni, 2021]

Inductive biases

Definition 1

An inductive bias is a **preference towards a certain type of generalization** over others. Multiple functions may fit the training data, the IB should ideally favour the ones that perform better OOD [Wilson and Frank, 2023]

Definition 2

An inductive bias frames **which continuation (OOD) is simpler** to learn for the learner. A learner is biased towards C_1 over C_2 if the training set and its extension according to C_1 has a **shorter description length** compared to C_2 [Kharitonov and Chaabouni, 2021]

Definition 3

Inductive biases are a **set of assumptions** that a learning algorithm employs to make predictions on unseen data based on its training data. The assumptions are **inherent in the algorithm's design** (<https://www.geeksforgeeks.org/machine-learning/what-is-inductive-bias-in-machine-learning/>)

Inductive biases

Definition 1

An i
othe

An
lear
acco

In
ma
ir



tain type of generalization over
a, the IB should ideally favour the
on and Frank, 2023]

For **IB^s**, we need a definition that is
general enough to be **extended to
simulators** but **precise** enough that
does not become useless

t a learning algorithm employs to
training data. The assumptions are
[www.geeksforgeeks.org/machine-](http://www.geeksforgeeks.org/machine-learning/what-is-inductive-bias-in-machine-learning/)

learning/what-is-inductive-bias-in-machine-learning/)

Inductive biases



Inductive biases

$$f_{GNN}(X, A) = C'(f_{GNN,1}(X_1, A_1), f_{GNN,2}(X_2, A_2), A_{new})$$

GNNs

Markovian

Mechanism invariance

Sequentiality

Weight sharing

Combinatorial

Relational

Translational equivariance

Counting

$$h_u^{k+1} = \Phi^k(h_u^k, \Theta \Psi^k(h_u^k, h_v^k, e_{uv}))$$

Transferability

Hierarchical structure

Simplicity bias

Long/short term memory

Locality

Factorization

Compositional

Translational invariance

Arithmetic

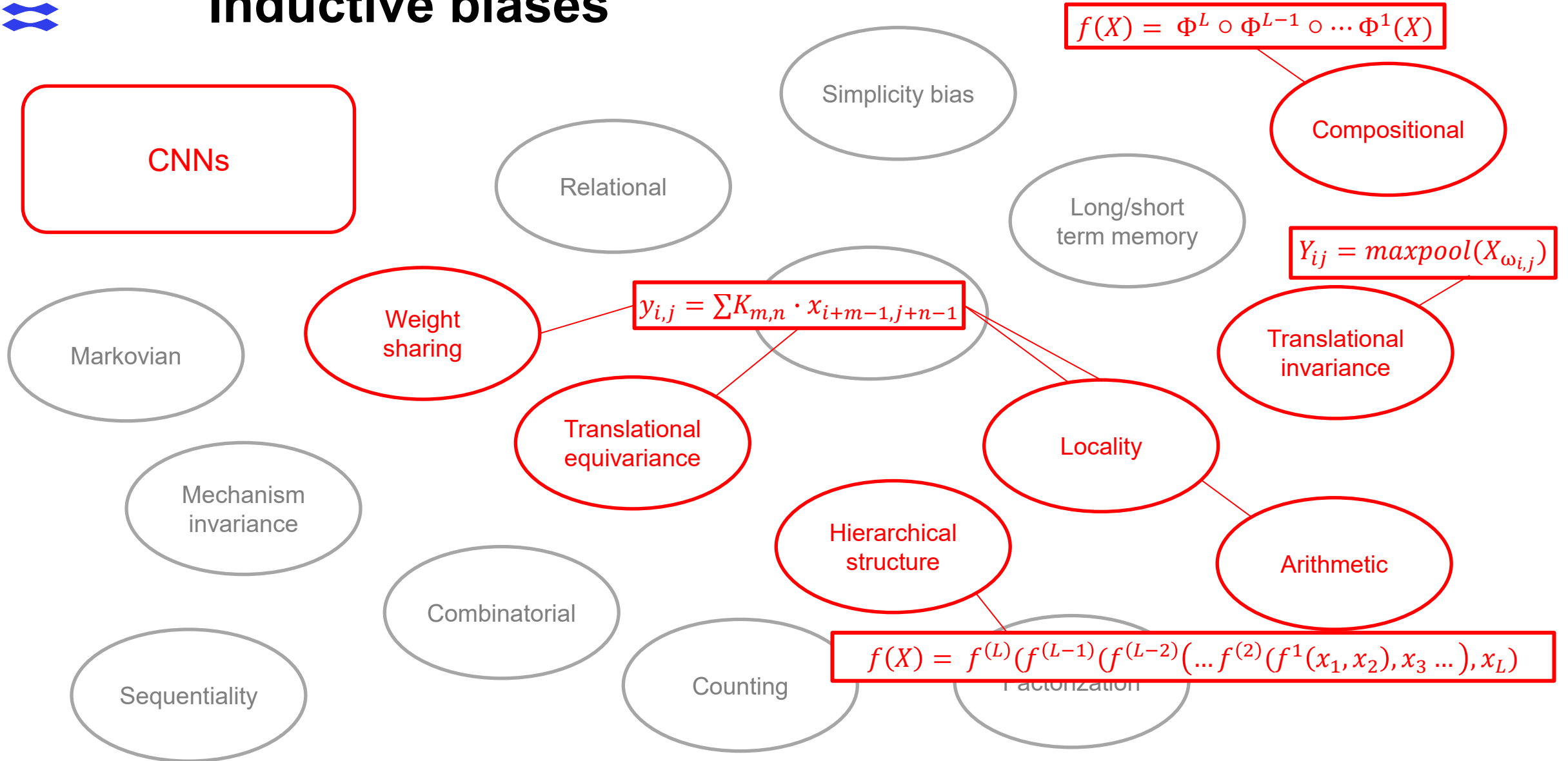
Permutation invariance

Ψ^k, Φ^k

h^{k+1}

Ψ^k

Inductive biases



Inductive biases

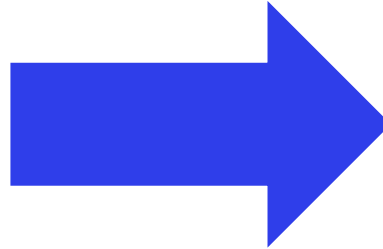
Review of 31 studies

1. **State-of-the-art NNs:** CNNs, GNNs, RNNs, Transformers, Multiplicative NNs, etc.
2. **Architecture features:** Layers, embeddings, physics informed, search space, hyperparameters, etc.
3. **Fields:** computer vision, language models, combinatorial problems, graph problems, algorithms, causality

Inductive biases from simulators

Review of 31 studies

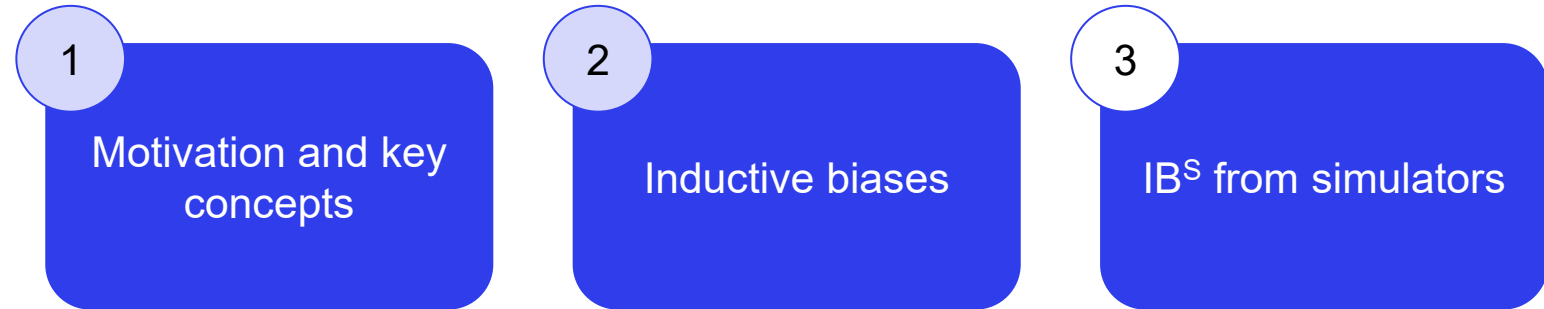
1. **State-of-the-art NNs:** CNNs, GNNs, RNNs, Transformers, Multiplicative NNs, etc.
2. **Architecture features:** Layers, embeddings, physics informed, search space, hyperparameters, etc.
3. **Fields:** computer vision, language models, combinatorial problems, graph problems, algorithms, causality



Operational definition for IB^S

*A potential IB^S is a **mathematically or algorithmically** embedded element of the simulation that **systematically** steers its results in a direction*

*Its formulation must be expressed in a **closed mathematical system** transferable to the NN into either the **architecture, parametrization, training space or input encoding***



Outline



Traffic simulations

Task
Shortest path calculation
Path selection/route choice
Edge cost estimation
Traffic assignment (network loading)

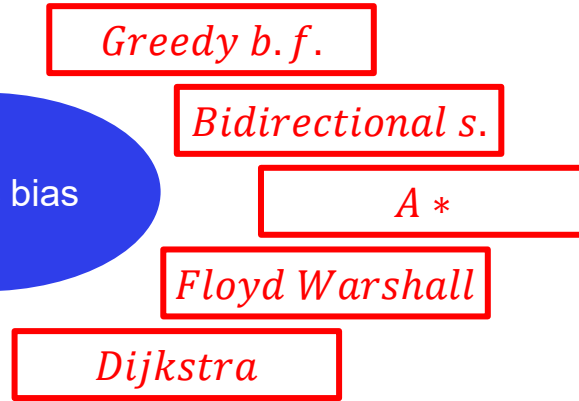
Traffic simulations

Task	Algorithm
Shortest path calculation	Dijkstra
	Bellman-Ford
	A*
	Floyd-Warshall
	Bidirectional search
	Greedy best-first search
Path selection/route choice	Logistic regression alg.
	Gawron
Edge cost estimation	Car-following alg.
	CTM
	Volume delay functions
Traffic assignment (network loading)	Frank Wolfe
	MSA
	SAP
	Simulation based assignment

From:

- Aimsun
- PTV VISUM
- SUMO
- TransCAD
- SimMobility
- MATSim
- Compass
- EMME
- DTALite
- Aequilibrae

Inductive biases



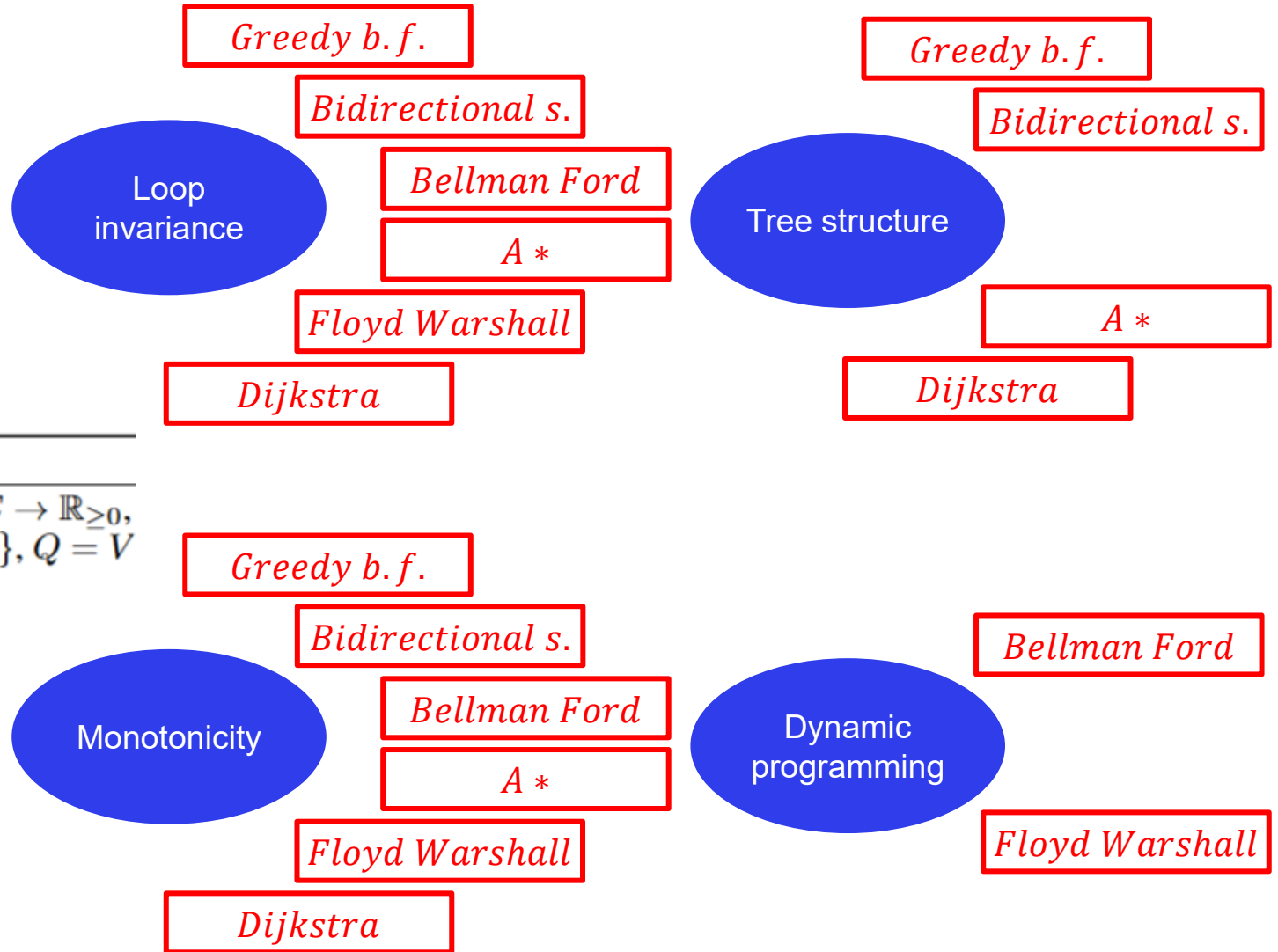
$$dist[A] + cost(A, B)$$

$$dist[A, B]' = dist[A, B] + dist[B] - dist[A]$$

- Positivity constraint
- Additive formulation of the cost
- Minimization of the objective

Inductive biases

IB^s



Algorithm 1 Dijkstra's Algorithm

Input: Weighted graph $G = (V, E, w)$ with $w : E \rightarrow \mathbb{R}_{\geq 0}$,

Initialize: $d(s) = 0$, $d(v) = \infty$ for all $v \in V \setminus \{s\}$, $Q = V$

while $Q \neq \emptyset$ **do**

$u \leftarrow \arg \min_{v \in Q} d(v)$

$Q \leftarrow Q \setminus \{u\}$

for each $(u, v) \in E$ **do**

if $d(u) + w(u, v) < d(v)$ **then**

$d(v) \leftarrow d(u) + w(u, v)$

$p(v) \leftarrow u$

end if

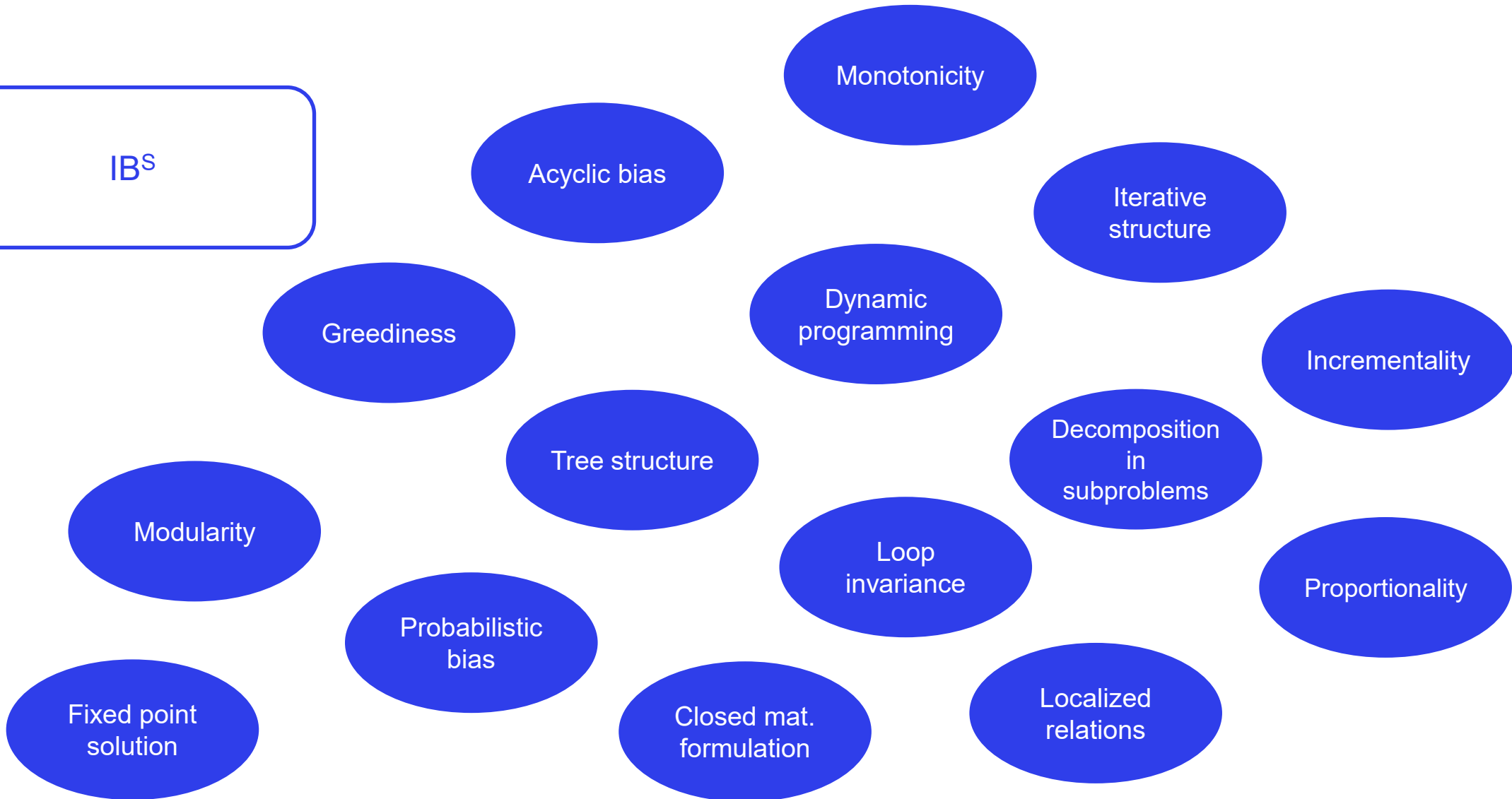
end for

end while

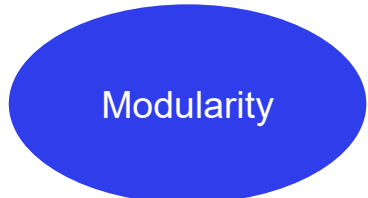
Output: Distances $d(v)$ and predecessors $p(v)$

Inductive biases

IB^s

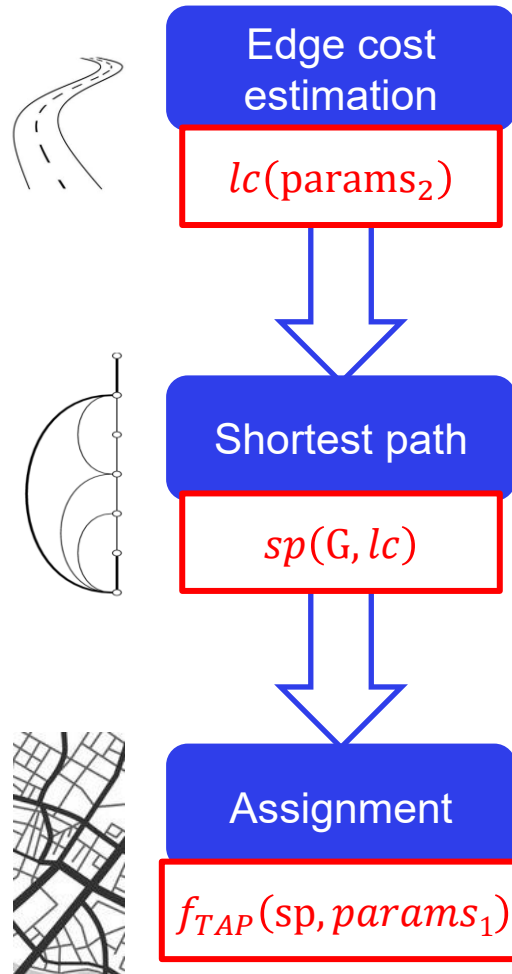


Inductive biases



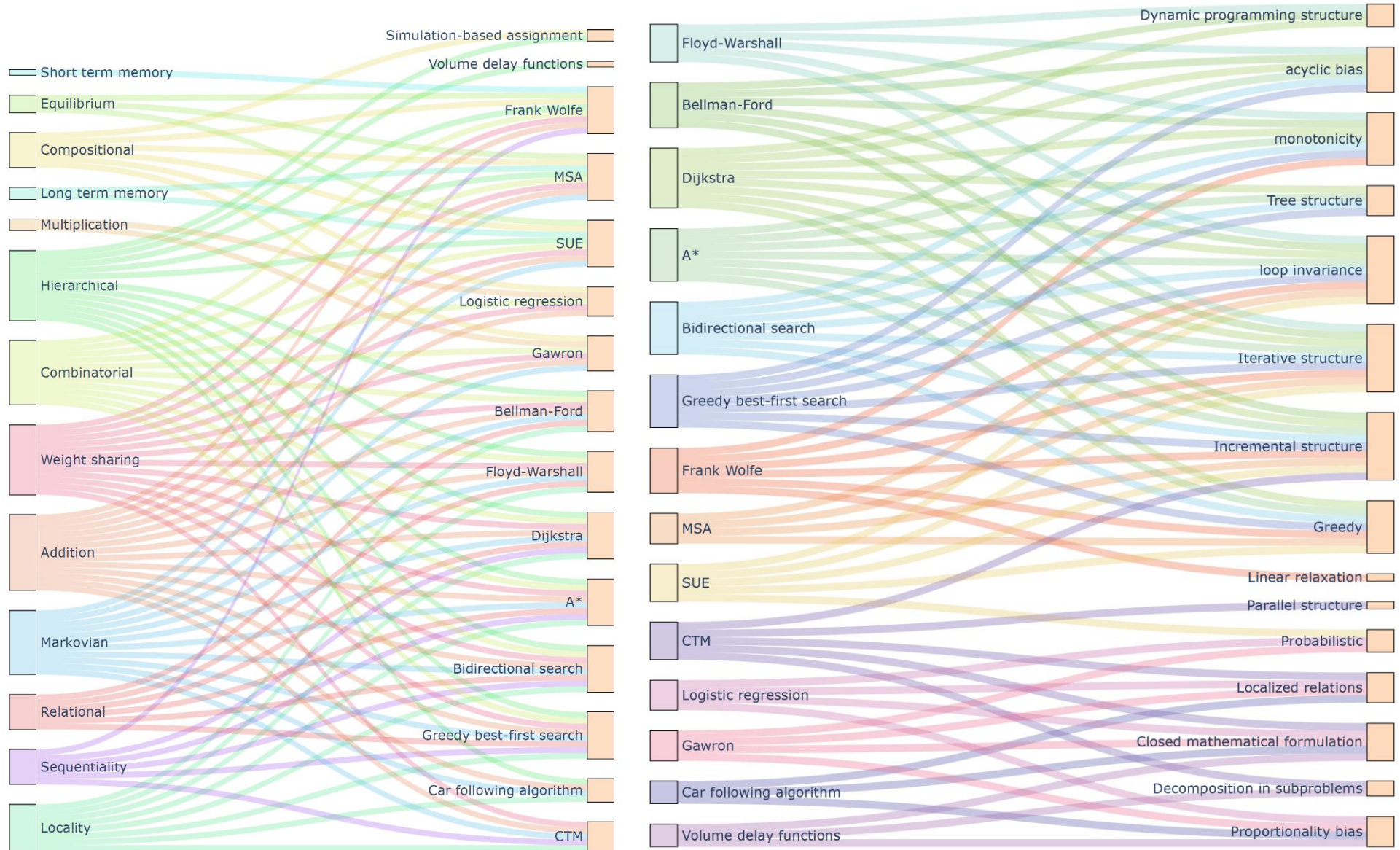
Hierarchical structure

$$F(SV) = f_{TAP}(sp(G, lc(params_2)), params_1)$$



- Localized relations
 - Closed mathematical formulation
 - Subproblem decomposition
 - Proportionality bias
-
- Dynamic programming
 - Acyclic bias
 - Monotonicity
 - Tree structure
 - Loop invariance
 - Iterative
 - Incremental
 - Greedy
-
- Monotonicity
 - Loop invariance
 - Iterative
 - Incremental
 - Greedy
 - Probabilistic
 - Linear relaxation

IB and traffic simulation tasks



C

Conclusions

Outline



Conclusions

1. **Literature is still too fragmented** to fully carry out **knowledge transfer** between specific fields and ad-hoc NN designs
2. **Simulators are a big source of “IB” and generalizing power**, the challenge is to **formalize** these in a way that is easily applicable
3. **Modularity is the biggest challenge** when trying to formulate simulators as **ensemble** of algorithms
4. **Embedding simulation features** theoretically tackles the two key challenges of **interpretability and robustness**

Thank you

Serio Agriesti; Francisco Camara Pereira
samaa@dtu.dk

DTU – Department of technology, management and economics