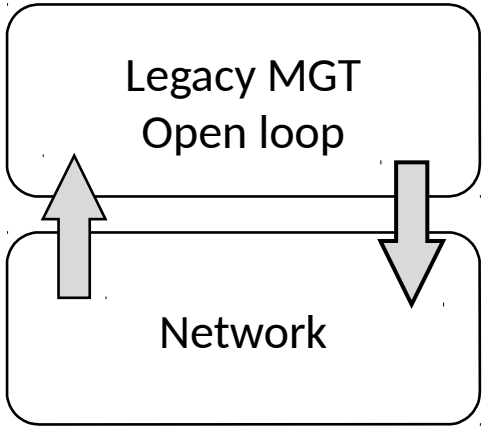
Beyond Cognitivism

IRTF NMRG Workshop on Network Intelligence

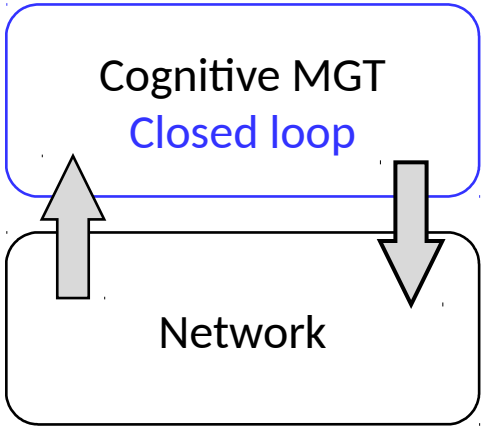
Dimitri Papadimitriou

Landscape

- **Legacy management**
- **Cognitive management**: automatically solve complex problems under uncertain (sometimes hostile) conditions to adjust/produce effective (re)action plans
  - Automaton: performs according to predetermined set of instructions
  - Complex: involve learning, inference/reasoning, causality analysis, expert knowledge, etc.
  - Uncertainty: epistemic (lack of knowledge) or aleatory (randomness/variability)
  - Closed loop: mainly adaptive (less often model-reference/predictive)



Closed system



Closed system

Mainly... combination of centralized "learning" with autonomic networking

Beyond cognitivism

Synthesis

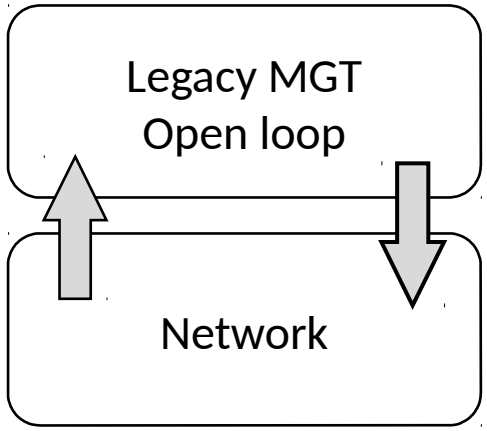
Questions	Behaviorism	Cognitivism
How does learning occur?	Black box - observable behavior main focus	Structured, computational
What factors influence learning?	Nature of reward, punishment, stimuli	Existing schema, previous experiences
What is the role of memory?	Memory is hardwiring of repeated experiences - where reward and punishment are most influential	Encoding, storage, retrieval
How does transfer occur?	Stimulus, response	Duplicating knowledge constructs of "knower"
What types of learning are best explained by this theory?	Task-based learning	**Reasoning, clear objectives, problem solving**

Source: Ireland, T. (2007). Situating connectivism. November 7, 2008

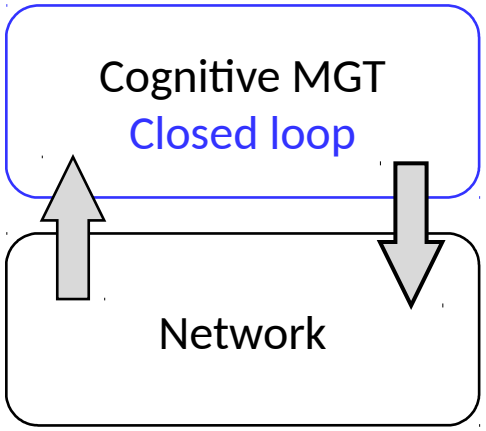
Synthesis between cognitivism and connectivism

- Cooperatively solve complex problems in various environments/under uncertain (sometimes hostile) conditions and autonomously decide/anticipate about course of actions
  - Autonomous: self-driving, self-governing … choice/alternatives
  - Complex: many hidden relations/dependencies between heterogeneous parts (H-M, M-M) … reconcile **common/shared** with **individual expectations**
  - Beyond Artificial Intelligence (master-slave) and Collective Intelligence (emergence)

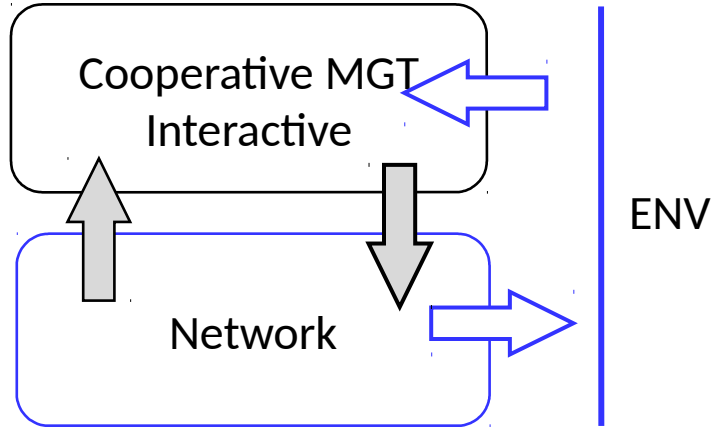
Yesterday: learning (ML/AI) to solve complex network problems
Tomorrow: network to solve complex learning problems = cooperative intelligence



Closed system



Closed system



Open system

Example

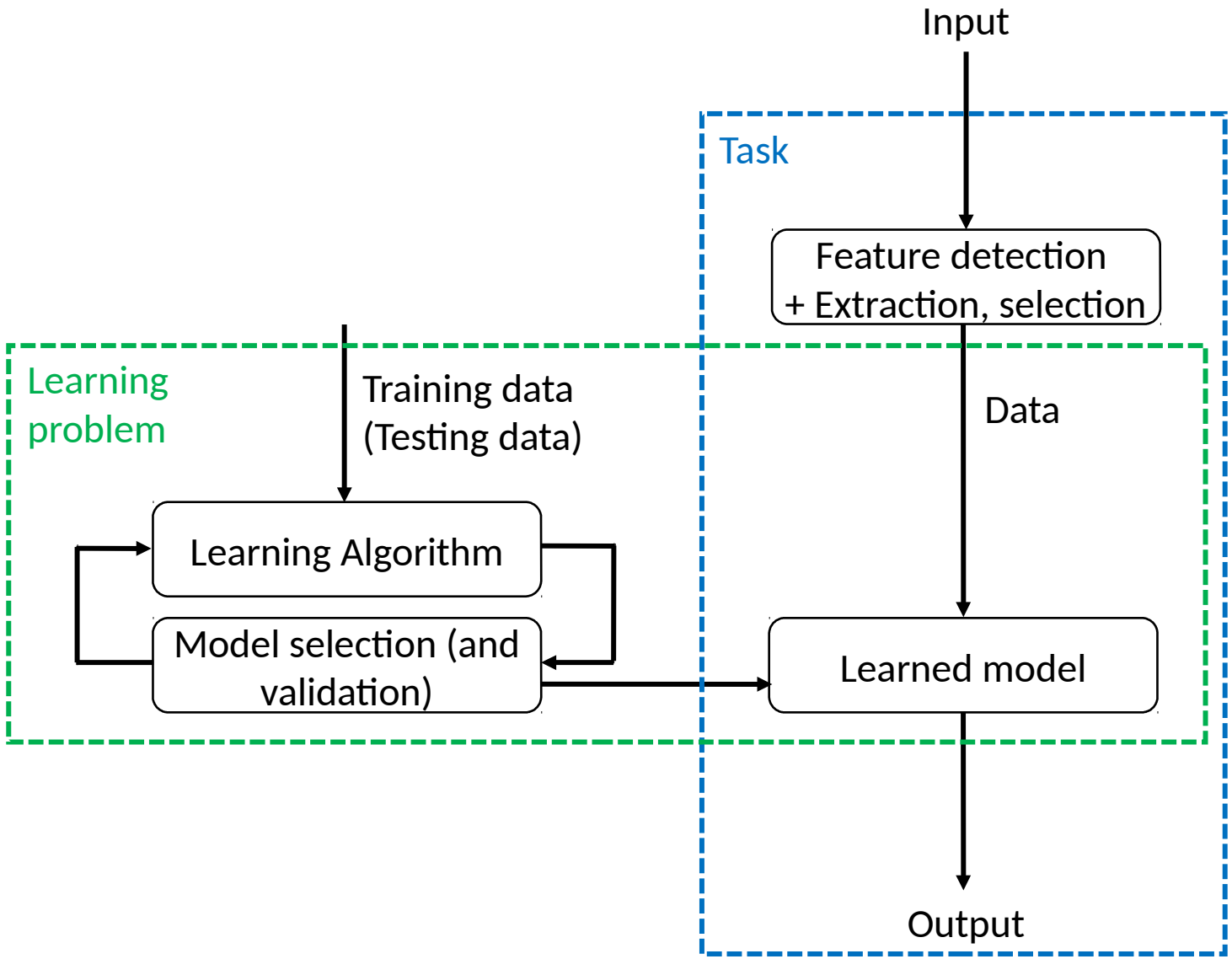
Principle: exploit "intelligence" to give up/delegate control to accountable parts … more can be achieved with less resources/effort (best use cases are not only those increasing utility/satisfaction of end-users but also their proactive involvement)

… good source of inspiration for use cases: DINRG

Key elements

- Data more than traffic, topology, etc. and objectives well beyond cost/performance improvement, fast reliable provisioning/dynamic resource allocation, service continuity, etc.
- Data sources and data processing not necessarily collocated in/owned by same admin. entity
- As part of cooperation, sharing data implies form of return/reward (not just "better" service)
- Contextual capability (together with situation, events, or information related to it, rather than alone)
- "Statistically reliable, and individually impressive" instead of "Statistically impressive, but individually unreliable"
- Prevent exploitation of flaws/weaknesses

Role of machine learning



Role of machine learning

- Machine learning can play a role (automation, prediction, etc.) … BUT
  - Production of learned model itself **not automated** - cf. automated ML research
    … Lots of external intervention/domain knowledge from collecting training data until
    ‖ obtaining learned model including learning algorithm
    - Example: classification task -> SVM or ANN ? … it depends
  - Difficult to process **heterogeneous**, **non i.i.d. data** (correlated, relational data), to operate under adversarial conditions (outlier detection, recent research in GAN), etc.
    ⇒ Know your data and applicability of out-of-the-shelve ML algorithms
    + a given "task" often involves feature extraction/selection and multiple learning problems
  - **Discriminative methods** (e.g. CRF, SVM, ANN) very limited "modeling" power compared to **Generative methods** (e.g., HMM, GMM) which build explicit model out of smaller data sets
    - Tradeoff: Performance (but less informative) vs. Flexibility (but more complex)
    - Example: determine the protocol corresponding to given trace
      - Generative: model protocols and determine to which protocol the trace belongs
      - Discriminative: model protocol differences (boundaries) without modeling any protocol

Exploitation

Setting experimental research objectives

- **Determine when and why "learning" is beneficial (cost/gain) and potentially harmful**
  - Evaluation beyond Alg. A performs better than Alg. B
  - ‖ Ex: network and end-user utility, gain, etc. but also tradeoffs (multi-obj. multi-period problems)
  - How to reconcile statistical (relational) learning with deterministic network operations
  - Repeatable (same or even different data sets), reproducible (other networks), verifiable
  - ‖ Aleatory uncertainty (characterize variance) ⇒ Generalization even harder (share learned models ?)
  - **Model**: bottom-up/peer exchanges (decentralized/organic) vs. top-down/master-slave (centralized)
  - ‖ Ex: multi-tenant/domain cases with partial vs. full information
- **Identify essential parts/components and relations/exchanges**
  - Modular chain from data sources (incl. H-H, H-M interactions) until output (+feedback)
  - **Problem solving**
    - Problem space very large (time/delay, energy/resources, complexity, etc.)
    - **Focus on finding commonality in solving methods, procedures, practices, etc.**
  - **Balance** between genericity/universality and specificity/ad-hoc
  - Languages beyond information model/data exchange protocols

Documentation

- **Lessons learned from "running" or previous projects**

  + some elements of common terminology, criteria and metrics, etc.

- **Overall finality and utility of "network intelligence" … common design goals**
  - Understanding of current management systems (models, procedures, etc.) and their limits
  - Design goals
    - Fundamental evolution from edges/periphery towards core
    - ‖ **or** incremental improvement network management functionality ?
    - Towards synthesis of cognitivism and connectionism
    - ‖ **or** automation of management procedures ?
      - Note: w.r.t. other ML domains: main diff. distribution/network dim. (since so far abundantly absent)
    - Mean to converge MGT levels
    - ‖ **or** keep isolation between distinct layers (e.g. PHY, TCP/IP, APP), partitions (mobile/fixed), etc.
  - Quantitative and qualitative criteria, metrics

- **Related datasets, tools and languages**
  - Tools … not limited to passive measurement (and related), e.g., shared libraries
  - Languages (beyond information model/data exchange protocols)