Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

CogTwin: A Hybrid Cognitive Architecture Framework for Adaptable and Cognitive Digital Twins

Sukanya Mandal[1] , **Noel E. O'Connor**[2]
[1]Research Ireland CRT in Machine Learning (ML-Labs), Dublin City University, Ireland
[2]Insight Research Ireland Centre for Data Analytics, Dublin City University, Ireland
sukanya.mandal2@mail.dcu.ie, noel.oconnor@dcu.ie

Abstract

Current Digital Twin (DT) technology lacks the cognitive capabilities needed for true autonomy and intelligent adaptation. This paper introduces CogTwin, a hybrid cognitive architecture framework for developing Cognitive Digital Twins (CDTs). CogTwin integrates a 50ms cognitive cycle inspired by human cognition, dual knowledge graphs (static Domain Knowledge Repository (DKR) and dynamic Internal Knowledge Graph (DIKG)), a hybrid attention mechanism, and self-healing capabilities. Combining symbolic, sub-symbolic, and neuro-symbolic AI, CogTwin enables real-time learning and decision-making. Simulated smart city scenarios, including traffic incident management and power outage response, demonstrate CogTwin's potential. Preliminary performance evaluations of the pseudocode suggest feasibility of the target 50ms cycle. The architecture also incorporates explainable AI (XAI) for transparency and human-CogTwin collaboration. CogTwin contributes towards a unified theory of cognition for DTs, laying the groundwork for more sophisticated and autonomous CDTs.

1 Introduction

Digital Twin (DT) technology is revolutionizing complex system management by creating virtual representations of physical entities and processes [Jones *et al.*, 2020]. While DTs often excel at real-time analysis and predictive modelling [Kobayashi and Alam, 2024], they often lack the cognitive capabilities necessary for true autonomy and intelligent adaptation in dynamic environments [Hribernik *et al.*, 2021]. Current DTs typically struggle to handle unforeseen events and complex, evolving situations due to their reliance on preprogrammed rules and data-driven models. This inability to learn, reason, and adapt in real time hinders the full realization of their potential.

CogTwin addresses this gap with a hybrid cognitive architecture (CogArchs) combining symbolic, sub-symbolic, and neuro-symbolic AI for robust reasoning and adaptive learning. CogTwin aims to imbue DTs with human-like cognitive abilities, enabling real-time learning, reasoning, and intelligent decision-making. Key features include a 50ms cognitive cycle inspired by human cognition; Dual Knowledge Graphs (KGs) - a static Domain Knowledge Repository (DKR) and a dynamic Internal Knowledge Graph (DIKG); a hybrid attention mechanism; and self-healing capabilities. Explainable AI (XAI) techniques ensure transparency and facilitate human-CogTwin collaboration. Initially, CogTwin's architecture inherently supports explainability through its symbolic components: the DKR and DIKG provide a structured representation of knowledge that can be queried to understand the basis for certain conclusions, and the rule-based systems in the Reactive Layer offer transparent decision logic. More sophisticated, dedicated XAI modules, such as those generating counterfactual explanations or visualizing attention weights in neural networks (as discussed in Section 5.3), are planned as future enhancements to interface with both symbolic reasoning paths and sub-symbolic model outputs, thereby providing deeper transparency into the hybrid system's operations. This research contributes a novel architectural framework (CogTwin available at[1]) and detailed pseudocode implementation for building CDTs.

2 Related Work

This section examines existing research relevant to CogTwin, encompassing CogArchs, DTs and Cyber Physical Systems (CPS), Landscape of CDTs, KGs, hybrid AI, attention mechanisms, real-time systems, and metacognition.

Existing CogArchs such as ACT-R [Anderson *et al.*, 2004], SOAR [Laird, 2019], LIDA [Franklin *et al.*, 2013], SIGMA [Rosenbloom, 2013] and CLARION [Sun, 2006] provides a foundation for cognitive modeling. However, they weren't initially designed for the real-time interaction and dynamic adaptation required by DT applications. While some have real-time extensions (e.g., ACT-R/E [Trafton *et al.*, 2013] and [Thórisson and Helgasson, 2012]), they typically rely on a single AI paradigm and face challenges in achieving rapid adaptation, complex system representation, and autonomous self-healing crucial for robust DT operation. CogTwin addresses these limitations with a hybrid AI approach within a unified framework. **Table: Comparison of**

[1]https://github.com/sukanyamandal/ProjectCogTwin

CogTwin with Cognitive Architectures [CogTwin, 2025d] provides a detailed comparison.

DT and CPS research [Singh *et al.*, 2021], [Jinzhi *et al.*, 2022], [Baheti and Gill, 2011] provides context for CogTwin. The field of CDTs [Zheng *et al.*, 2022] is nascent, primarily limited to exploring theoretical frameworks and applications. CogTwin offers a holistic CogArch framework for DT integration.

CogTwin in the **landscape of CDTs** [Shahzad *et al.*,]; building upon foundational CogArchs (detailed in [CogTwin, 2025d], carves a distinct niche by specifically addressing the unique demands of DTs. To clearly position CogTwin and highlight its advancements within this emerging domain, **Table: Comparison of CogTwin with Representative Cognitive Digital Twin (CDT) Architectures [CogTwin, 2025e]** provides a comparative analysis against representative contemporary CDT frameworks [Calderita *et al.*, 2020], [Lv *et al.*, 2023], [Abburu *et al.*, 2020], [Lu *et al.*, 2020], [Eirinakis *et al.*, 2020], [Du *et al.*, 2020]. This comparison covers key dimensions including architecture type, cognitive capabilities, knowledge representation, and real-time responsiveness.

KGs are crucial for CogTwin's knowledge representation (KR) and reasoning. Research in KG construction, reasoning, and application informs CogTwin's dual KG approach (static DKR and dynamic DIKG), enabling the representation of both stable domain knowledge and dynamic, real-time information. The DKR addresses real-time KG reasoning challenges in dynamic DT environments.

CogTwin's **hybrid AI** [Kunz *et al.*, 1984] architecture draws upon research in symbolic [Smolensky, 1987], sub-symbolic [Ilkou and Koutraki, 2020], and neuro-symbolic AI [Bhuyan *et al.*, 2024]. It leverages each paradigm's strengths: symbolic for explainable reasoning, sub-symbolic for learning and adaptation, and neuro-symbolic for bridging the gap. Research on KR [Prasad and others, 2012] and reasoning techniques [Bettini *et al.*, 2010] further informs CogTwin's self-healing and XAI capabilities.

CogTwin's hybrid **attention mechanism** (self-attention & cross-attention [Vaswani, 2017] dynamically prioritizes information, enabling the 50ms cognitive cycle.

CogTwin's 50ms cognitive cycle is based on research on **real-time and reactive systems** [Stankovic and others, 1988], [Harel and Pnueli, 1984] incorporating principles of real-time scheduling [Sha *et al.*, 2004], event-driven architectures [Michelson, 2006], and timing analysis [Liu *et al.*, 2025]. This rapid cycle is essential for closed-loop interaction and control within dynamic DT environments.

CogTwin's **metacognitive layer**, enabling self-monitoring, self-regulation, and self-healing [S-Julián *et al.*, 2023], draws upon research on metacognitive architectures [Samsonovich, 2009] and computational models of metacognition [Cox, 2011]. This layer contributes to robustness and resilience in dynamic environments.

3 CogTwin: A Hybrid CogArch Framework for Adaptable and Cognitive DTs

Current DT technologies, while effective for managing complex systems, lack the cognitive capabilities needed for true autonomy and intelligent adaptation. This limitation motivates the development of CDTs, enhancing DTs with advanced reasoning, learning, and decision-making capabilities. CogTwin, a hybrid cognitive architecture, addresses this gap by providing a framework for realizing CDTs in dynamic environments (formalized in **CogTwin Framework** [CogTwin, 2025b]). CogTwin's design, guided by key principles from [Newell, 1994], aims to integrate diverse cognitive functions within a practical framework tailored for the DT domain. This section details CogTwin's modular architecture and its target 50ms cognitive cycle for real-time responsiveness.

3.1 Architecture Overview and Framework Approach

CogTwin integrates symbolic, sub-symbolic & neuro-symbolic AI paradigms to overcome the limitations of single-paradigm approaches for CDTs. This hybrid approach is crucial for creating CDTs capable of both robust, explainable reasoning and powerful, adaptive learning. Symbolic AI (using the DKR and DIKG) provides KR and logical reasoning. Sub-symbolic AI (through neural networks (NNs) in the Deliberative Layer) enables learning complex patterns. Neuro-symbolic integration combines the strengths of both, essential for real-time responsiveness and intelligent decision-making in dynamic DT environments. This framework approach offers modularity, extensibility, standardization, and interoperability. CogTwin's target 50ms cognitive cycle, inspired by human cognition [Newell, 1992], enables real-time interaction. The following subsections detail the individual modules and their interactions within this cycle.

3.2 Modules, Integration, and Workflow

CogTwin comprises the following key modules designed to interact within a continuous cognitive cycle, aiming for real-time responsiveness and adaptation within the DT environment (see **Table: CogTwin Module Interactions** [CogTwin, 2025c] for module interactions):

DKR Layer (World Model KG): The DKR provides the initial static world knowledge. Constructed offline using methods from [Mandal and O'Connor, 2024], [Mandal and O'Connor, 2024] for multimodal KG and ontology creation, it encapsulates the DT environment's representation. It is periodically updated offline with new knowledge gained through CogTwin's interactions.

Perceptual Buffers: Bridging the DT environment and CogTwin, these buffers receive real-time sensor data and knowledge injections. An integrated hybrid attention mechanism (self- and cross-attention [Vaswani, 2017]) filters and prioritizes relevant data by dynamically balancing focus on internal context (self-attention) with correlating information across streams or with DKR/DIKG knowledge (cross-attention). This balance is governed by cognitive goals, data salience, and learned policies optimizing information throughput.

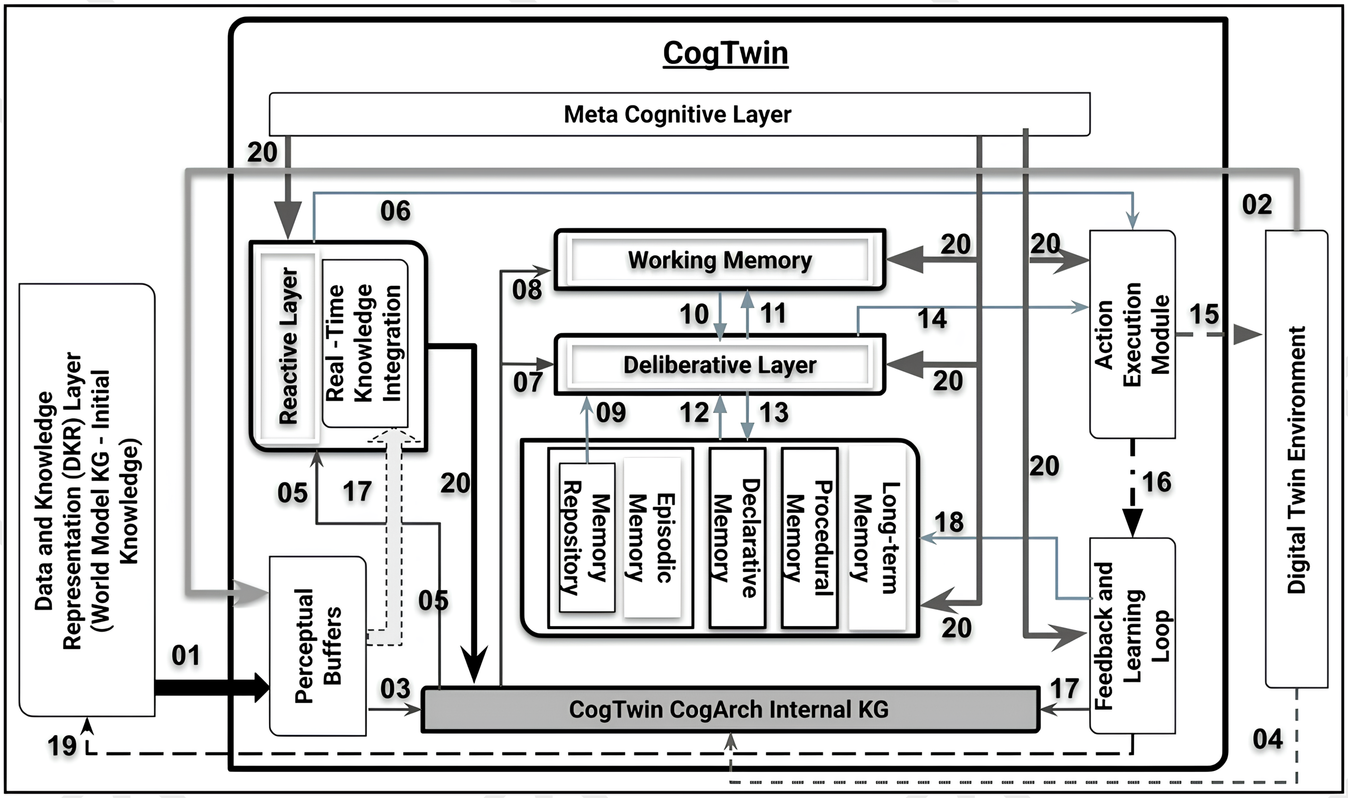

Figure 1: CogTwin: A Hybrid Cognitive Architecture Framework for Adaptable and Cognitive Digital Twins

Real-Time Knowledge Integration (RTKI) Module: The RTKI processes filtered perceptual data, dynamically integrating it into CogTwin's DIKG. It transforms raw data into symbolic representations compatible with the DKR's ontology, resolving inconsistencies and managing conflicts using ontological rules and confidence scores (e.g., prioritizing newer or higher-confidence data, or cross-referencing related points). Complex/persistent inconsistencies are flagged for Meta-Cognitive Layer assessment.

CogTwin DIKG: This dynamic KG is CogTwin's central hub, storing the perceived and interpreted state of the DT environment. Continuously updated by the RTKI, the DIKG provides context for all cognitive functions and serves as input to NNs. A simplified formal data flow representation can be found in **CogTwin Data Flow: Perceptual Buffer to Internal KG Update** [CogTwin, 2025a]. Future work will explore a more complete formalization using OWL.

Reactive Layer: Operating rapidly, the Reactive Layer handles immediate actions based on DIKG patterns. Its rule-based system (e.g., IF <condition> THEN <action>), with rules stored in Procedural Memory, ensures swift responses to critical situations. For example, IF (DIKG contains ``Power Outage'' AND ``Hospital'' is affected) THEN (initiate emergency power).

Working Memory: Acts as a short-term memory store, holding information relevant to CogTwin's active goals and tasks. It extracts this information from the DIKG based on the current context and provides a workspace for the Deliberative Layer, holding both symbolic and sub-symbolic data.

Deliberative Layer: Responsible for higher-level cognitive functions (planning, decision-making, complex reasoning). It interacts with Working Memory, the DIKG, and LTM, employing a hybrid reasoning approach combining various reasoning types (case-based, deductive, inductive, abductive, analogical, temporal, and spatial). Integrated NNs (e.g., GNNs) operate on the DIKG to learn and refine reasoning strategies.

Long-Term Memory (LTM): The LTM stores learned knowledge and experiences in a hybrid format (symbolic and sub-symbolic), comprising Declarative, Procedural, and Episodic Memory (including the Memory Repository for Case-Based Reasoning).

Memory Repository (within Episodic Memory): Enables Case-Based Reasoning (CBR) within the Deliberative Layer by storing past experiences as cases for retrieval and adaptation to new situations.

Action Execution Module: Translates decisions from the Reactive or Deliberative Layers into actions within the DT environment via a REST API. It also receives feedback from the environment on action outcomes.

Feedback and Learning (F&L) Loop: Closes the cognitive cycle by using action outcomes and feedback to update the DIKG, LTM, and periodically the DKR. It incorporates learning algorithms [Park and Park, 2020] and provides feedback to NN components.

Meta-Cognitive Layer: Monitors and controls the overall cognitive process, assessing performance, detecting inconsistencies, and dynamically allocating resources for robust and efficient operation.

3.3 DKR Enrichment and Evolution

The DKR serves as a static baseline during real-time operation but is periodically enriched offline to incorporate new knowledge, balancing stability with long-term learning. (See **CogTwin Framework** [CogTwin, 2025b] for details). The enrichment process involves:

Initial DKR Construction: The DKR is initially constructed offline using two complementary approaches: (a) real-world data processing [Mandal and O'Connor, 2024] to create an initial ontology and populate the DKR with factual information sourced from the specific DT environment, and (b) a synthetic multimodal knowledge graph (MMKG) generation method [Mandal and O'Connor, 2024] to address potential sparsity issues and provide a rich initial knowledge base, particularly crucial during development phases; production environments would ideally leverage comprehensive real-world data from the outset. This combined approach ensures a robust and comprehensive starting point for CogTwin's cognitive processes.

CogTwin Operation and Learning: CogTwin interacts with the DT environment, updating its DIKG through real-time data processing and learning. The DKR remains unchanged during this phase, providing a stable knowledge base for real-time operations within the 50ms cycle. The RTKI module dynamically processes real-time data to update the DIKG, and the F&L Loop updates the DIKG based on learned experiences.

Offline DKR Enrichment (Periodic): Learned information deemed valuable for long-term knowledge is extracted from CogTwin's DIKG.

This extracted knowledge forms the primary basis for offline DKR updates. The DKR is enriched by: (a) first, integrating this consolidated valuable knowledge from the DIKG. (b) If knowledge gaps are identified after this step (primarily during development, as production ideally relies on complete real-world data), synthetic data is then generated using the [Mandal and O'Connor, 2024] method to address these specific gaps. The complete set of enriched knowledge (DIKG-derived and any supplementary synthetic data) is subsequently integrated into the DKR using the ontology-based approach [Mandal and O'Connor, 2024], ensuring overall consistency and structure.

This offline enrichment enables intensive validation. Conflicts between DIKG-derived knowledge and the DKR are managed by enforcing consistency with the established ontology (including schema validation, semantic contradiction checks, and prioritization of recent/validated data). Further safeguards against corruption include ontological alignment, DKR versioning, and human review for complex or conflicting integrations.

This offline update enables intensive knowledge consolidation without impacting real-time operations. DKR enrichment is triggered by schedules, significant new DIKG knowledge accumulation, or detected performance degradation. 'Valuable' DIKG information is selected for DKR incorporation based on its stability, persistence, application success, novelty, added value, and goal impact. This semi-automated selection, using F&L and DIKG metrics, may include human oversight for critical integrations to ensure accuracy.

Next Cognitive Cycle: The updated DKR serves as the initial knowledge base for the next operational cycle of CogTwin. This iterative process of DKR enrichment allows CogTwin to continuously learn and adapt to long-term changes in the DT environment.

3.4 The 50ms Cognitive Cycle

CogTwin is designed to operate on a 50ms cognitive cycle for rapid responses and continuous learning, inspired by models of human cognition and cognitive architectures [Anderson *et al.*, 2004], [Newell, 1992], [Just and Carpenter, 1992], [Kieras and Meyer, 1997]. This cycle comprises the following phases (full pseudocode in [CogTwin, 2025b]):

Perception (Target: 5ms): Perceptual Buffers receive sensory data and textual information. An attention mechanism filters and prioritizes information. Relevant DKR portions are accessed for context.

Knowledge Integration (Target: 10ms): The RTKI module integrates filtered data into the DIKG, transforming raw data into symbolic representations consistent with the DKR ontology.

Situation Assessment (Target: 10ms): The Deliberative Layer assesses the situation by analyzing the DIKG, Working Memory, and potentially the Memory Repository and Long-Term Memory. NNs may contribute to this assessment.

Planning and Decision-Making (Target: 15ms): The Deliberative Layer formulates a plan, potentially using search algorithms, heuristics, or NN input. For simpler actions, the Reactive Layer may select an action directly.

Action Selection and Execution (Target: 5ms): The Action Execution Module selects and executes the chosen action, sending commands to the DT environment.

F&L (Target: 5ms): The F&L Loop processes feedback from the DT environment, updating the DIKG and Long-Term Memory. Periodically, the DKR is updated offline. Feedback is also provided to NNs, refining their learning process.

Metacognitive Monitoring and Control (Continuous - Low Overhead): Throughout the entire cycle, the Meta-Cognitive Layer monitors the performance of all modules, including NNs. It detects inconsistencies, adapts strategies, and dynamically allocates computational resources to optimize performance and ensure robust operation. This monitoring should have minimal overhead to avoid impacting the 50ms cycle. To ensure this minimal overhead, the Meta-Cognitive Layer primarily employs lightweight mechanisms such as sampling-based monitoring of key performance indicators rather than exhaustive checks of all processes, event-driven triggers for deeper analysis (e.g., when performance metrics drop below a predefined threshold or inconsistencies are repeatedly flagged by other modules), and efficient heuristics for performance assessment. More computationally intensive meta-cognitive processes, such as detailed strategy re-evaluation, would be scheduled to run with lower priority or during periods of lower cognitive load, thus preserving the integrity of the real-time cycle.

This 50ms target cycle enables CogTwin to adapt to the DT environment and learn from experience. Future work will focus on optimizing these timings for real-world deployments.

3.5 Integration of Cognition

CogTwin's architecture integrates key cognitive principles to create a unified system within the DT domain:

Dual-Process Theory: CogTwin, inspired by the dual-process theory of cognition [Kahneman, 2011], integrates two processing modes: a fast, intuitive System 1 (Perceptual Buffers, RTKI, DIKG) for rapid responses, and a slower, analytical System 2 (Deliberative Layer) for complex reasoning. Example: In traffic management, System 1 could adjust traffic lights locally, while System 2 optimizes overall traffic flow.

Cognitive Control: Action selection integrates symbolic reasoning (Deliberative Layer), sub-symbolic learning (NNs), and Reactive Layer inputs. The Meta-Cognitive Layer monitors performance and adjusts strategies. Example: If the Reactive Layer performs poorly, the Meta-Cognitive Layer might shift control to the Deliberative Layer.

Lifelong Learning: Memory Modules and the F&L Loop enable lifelong learning. The Memory Repository stores past experiences, while Long-Term Memory retains learned knowledge. Example: In predictive maintenance, CogTwin could learn to anticipate equipment failures by analyzing historical data.

Goal-Oriented Behavior: CogTwin operates in a goal-directed manner, with goals provided externally or derived internally. All cognitive processes are directed towards achieving these goals. Example: In smart grid management, a goal could be minimizing energy consumption while maintaining reliability.

3.6 Data Handling and Distributed Cognition

CogTwin incorporates strategies for efficient data handling and scalability in complex DT environments:

KG Compression and Optimization: Techniques like graph summarization [Liu *et al.*, 2018], indexing [Zhao *et al.*, 2007], and partitioning [Buluç *et al.*, 2016] reduce the computational cost of reasoning and retrieval. Example: Graph summarization can create a compact DKR for faster access during perception.

Attention Mechanisms: Attention mechanisms in the Perceptual Buffers prioritize relevant information, filtering noise and irrelevant details. These can be goal-directed or salience-based. Example: In security applications, attention mechanisms could prioritize video feeds showing unusual activity.

Distributed Processing and Cognition: CogTwin's modular design supports distributed processing and future integration with multi-agent systems (MAS) [Sycara, 1998]. Multiple CogTwin instances could collaborate, specializing in different aspects of the application. Example: In a smart city, different CogTwin instances could manage individual districts, coordinating actions for city-wide optimization.

These strategies, inspired by Newell's criteria for unified theories of cognition [Newell, 1994], aim to provide a robust and efficient CogArch for the DT domain.

4 CogTwin Use Case Scenarios: Smart City Applications

This section presents four different smart city scenarios illustrating CogTwin's potential for unified cognition. These use cases are further detailed in pseudocodes [CogTwin, 2025f], [CogTwin, 2025g], [CogTwin, 2025h], [CogTwin, 2025i], [CogTwin, 2025j] demonstrating the interaction between CogTwin's modules and the 50ms cycle.

4.1 Use Case 1: Traffic Incident Management

Scenario Description: A multi-vehicle accident occurs during rush hour at a major intersection, causing significant traffic congestion. This scenario involves complex interactions between vehicles, traffic signals, the road network, and emergency services.

CogTwin's Approach: CogTwin leverages its hybrid cognitive architecture to address this challenge. Real-time data from traffic cameras, GPS devices, and social media feeds populate the Perceptual Buffers. An attention mechanism prioritizes information related to the incident. The RTKI module integrates this filtered data into the dynamic DIKG, representing the current traffic state. The static DKR provides context, including knowledge of the road network, traffic light control logic, and emergency service protocols. Within the 50ms cycle, CogTwin's Deliberative Layer reasons about the incident's impact, predicting congestion propagation and evaluating alternative routes based on pre-defined rules and learned patterns. The Reactive Layer can quickly implement immediate traffic light adjustments at the affected intersection. The planned actions are then executed, dynamically adjusting traffic light timings, recommending alternative routes via navigation apps, and notifying emergency services. (See **Use Case 1: Traffic Incident Management** [CogTwin, 2025f])

Expected Outcomes: CogTwin's real-time adaptation and reasoning capabilities can mitigate congestion more effectively than traditional systems, resulting in reduced travel times, minimized disruption, and improved emergency response times.

Future Implications: This use case highlights CogTwin's potential to transform traffic management. Future research will explore incorporating predictive models and leveraging machine learning on the KG to anticipate congestion and implement preventative measures.

4.2 Use Case 2: Power Outage Response

Scenario Description: A transformer failure causes a power outage in a residential area during extreme heat, presenting challenges in managing the power grid, ensuring critical infrastructure operation, and addressing vulnerable populations' needs.

CogTwin's Approach: CogTwin integrates data from smart meters, social media sentiment analysis, and weather information into its DIKG. The DKR provides context about the power grid topology, critical infrastructure, and demographics, including vulnerable residents. CogTwin's hybrid reasoning assesses the outage's impact, prioritizes power restoration to critical facilities and vulnerable populations, and evaluates strategies for rerouting power and deploying backup generators. The 50ms cycle allows for rapid response and adaptation. (See **Use Case 2: Power Outage Response** [CogTwin, 2025g])

Expected Outcomes: CogTwin reduces outage duration and minimizes the impact on critical infrastructure and vulnerable populations by optimizing power rerouting, activating backup generators, dispatching repair crews, and providing timely information to residents.

Future Implications: This use case illustrates CogTwin's potential for proactive and adaptive grid management. Future research will explore incorporating predictive models to anticipate equipment failures and optimize preventative maintenance.

4.3 Use Case 3: Smart Home Automation

Scenario Description: A resident returns home on a hot day. The smart home system must optimize comfort and energy efficiency based on the resident's preferences, current conditions, and historical energy usage.

CogTwin's Approach: CogTwin integrates data from the smart home system (presence sensors, smart thermostats, energy readings) into its DIKG. The DKR contains information about resident preferences and historical energy profiles. CogTwin's reasoning determines optimal settings for devices (thermostats, lighting, appliances), considering comfort, cost, and environmental impact. CogTwin learns and adapts to evolving preferences. The 50ms cycle ensures rapid adjustments. (See **Use Case 3: Smart Home Automation** [CogTwin, 2025h])

Expected Outcomes: CogTwin offers a more adaptive and efficient approach to smart home automation, personalizing the environment and maximizing comfort while minimizing energy consumption.

Future Implications: This use case demonstrates CogTwin's potential for personalized and adaptive smart homes. Future research will explore incorporating more sophisticated learning algorithms to anticipate resident needs and proactively adjust settings.

4.4 Use Case 4: Medical Emergency Response

Scenario Description: A person experiences a medical emergency. Rapidly dispatching the nearest available ambulance and alerting the appropriate medical team is critical.

CogTwin's Approach: CogTwin receives real-time data about the emergency (location, type, vital signs). The DKR provides information about ambulance locations, hospital capacities, and medical team expertise. CogTwin's reasoning determines the optimal response, considering severity, resource proximity, hospital capacity, and patient needs. The real-time cycle ensures rapid dispatch and resource allocation. (See **Use Case 4: Medical Emergency Response** [CogTwin, 2025i])

Expected Outcomes: CogTwin reduces response times, improves patient outcomes, and optimizes resource utilization by rapidly dispatching ambulances, alerting medical teams, and providing real-time information to first responders.

Future Implications: This use case illustrates CogTwin's potential to transform emergency response. Future research will explore predictive models to anticipate demand and optimize resource allocation. Integration with the traffic management system (Use Case 1) will enable dynamic ambulance routing to minimize congestion-related delays.

4.5 Use Case: Cross-Domain Interaction

Smart city systems are interconnected, allowing CogTwin to demonstrate unified cognition by managing cross-domain interactions. For example, a traffic incident (Use Case 1) can impact ambulance response times (Use Case 4). CogTwin can prioritize emergency vehicle routes by adjusting traffic signals. Similarly, a power outage (Use Case 2) can affect traffic lights and smart homes (Use Case 3). CogTwin leverages its KG and hybrid reasoning to coordinate responses across these domains, demonstrating its potential for integrated smart city management. (See **Use Case: Cross-Domain Interaction** [CogTwin, 2025j])

5 Discussion and Future Work

CogTwin presents a novel hybrid CogArch for CDTs, enhancing DT autonomy and adaptability. Key characteristics include the dual KG (DKR and DIKG), the 50ms cognitive cycle, the hybrid attention mechanism, and self-healing capabilities. Integrating symbolic, sub-symbolic, and neuro-symbolic AI, CogTwin combines robust reasoning with powerful learning. The smart city use cases demonstrate its generalizability and capacity for cross-domain reasoning and dynamic adaptation - showcasing the realization of unified theory of cognition.

5.1 Performance Evaluation and Feasibility Analysis

A core requirement is the 50ms cognitive cycle. While full empirical evaluation awaits implementation, analysis using Big O notation and estimated execution times based on realistic hardware, software, and data volume assumptions suggest feasibility. Preliminary estimates indicate the 50ms target is achievable on a cloud server across all use cases. However, achieving this target on a Raspberry Pi 4 presents challenges, particularly for computationally intensive tasks like traffic management. Mitigation strategies include hierarchical graphs, pre-computed paths, and optimized graph libraries (See **Use Cases: Performance Evaluation** [CogTwin, 2025k]).

5.2 Current Implementation and Next Steps

The associated pseudocodes provides a detailed blueprint for future software implementations. The next major step is translating this pseudocode into a working implementation, followed by rigorous testing in simulated and real-world settings. Future work will prioritize empirical validation on target hardware to refine estimations and confirm the 50ms target feasibility, especially on resource-constrained devices. This validation is crucial for demonstrating CogTwin's real-time capabilities. Further research will explore more sophisticated learning algorithms, predictive modeling for proactive adaptation, and expanded cross-domain interaction capabilities.

5.3 Future Directions

Building upon the current pseudocode framework and simulated scenarios, future research will focus on the following key areas:

Functional Software Implementation and Real-World Validation: Our immediate next step is translating the pseudocode into a functional software implementation. Following implementation, rigorous testing and validation will be conducted in both simulated and real-world smart city pilot projects. Key Research Questions: What are the optimal software design patterns for real-time DT environments? How can we manage the computational demands of neuro-symbolic components and KG operations at scale?

Performance Optimization, Sensitivity Analysis and Scalability: We will conduct detailed performance evaluations, including sensitivity analysis and optimization. Crucially, this involves identifying/developing suitable benchmarks for CogTwin's cognitive abilities, addressing a current gap in standardized CDT evaluation that frameworks like CogTwin can help fill. Scalability testing will assess large-scale applicability. Key Research Questions: Effective real-time optimization strategies? Dynamic adaptation of strategies? CogTwin's scalability limits?

Unified Frameworks for CDT Standards and Benchmarks: The lack of CDT benchmarks is linked to the absence of standardized methodologies or architectures for cognitive capabilities in DTs; nascent CDT development features diverse, bespoke approaches hindering comparison and interoperability [Khan *et al.*, 2023]. Unified frameworks like CogTwin, with its structured architecture, can serve as a blueprint, facilitating standardized CDT development methods, interfaces, and performance metrics. This is essential for accelerating benchmark creation, fostering cohesive research, and enabling broader adoption and interoperability of advanced CDT systems.

KG Refinement and Automated Ontology Learning: Refining KR within CogTwin involves developing more expressive ontologies and automated learning methods for populating and updating the DKR and DIKG. Key Research Questions: How can we ensure consistency between the DKR and DIKG? How can we optimize KG representations for efficient reasoning and retrieval? What are the trade-offs between completeness and computational complexity?

Advanced Reasoning and Learning: Integrating probabilistic reasoning, deep reinforcement learning, and XAI will enhance CogTwin's capabilities [Parnafes and Disessa, 2004], [Furbach *et al.*, 2019], [Anshakov and Gergely, 2010]. Research will focus on dynamically selecting reasoning methods based on context. Key Research Questions: What are the best strategies for dynamically selecting reasoning and learning methods? How can we evaluate the effectiveness of different approaches in real-world scenarios? How can we leverage learned knowledge to improve future decision-making?

Human-CogTwin Collaboration: Designing intuitive interfaces [Rogers, 2012] for human operators to monitor, provide feedback, and intervene will foster collaboration. This involves developing mechanisms for explaining CogTwin's reasoning. Key Research Questions: How can we build trust and transparency in human-CogTwin interactions? How can XAI address ethical concerns related to transparency and bias? How can we balance automation and human control?

Enhancing Cognition with Theory of Mind (ToM) and Information Theory (IT): Incorporating ToM [Byom and Mutlu, 2013] will enable CogTwin to reason about other agents' beliefs and intentions. IT [Ash, 2012] can optimize information flow for more efficient decision-making. Key Research Questions: How can ToM and IT be integrated with other cognitive modules? What are the computational challenges of incorporating ToM and IT in real-time? How can we evaluate the benefits of ToM and IT in multi-agent scenarios?

Multi-Agent Cognitive Systems (MAS): Integrating MAS [Van der Hoek and Wooldridge, 2008] principles will enable multiple CogTwin instances to collaborate and coordinate, enhancing scalability and management of urban systems. Key Research Questions: How can we manage conflicts and ensure consistent behavior in a MAS? How can we distribute cognitive tasks among agents? What are the challenges of coordinating actions in a decentralized system?

Federated Learning and Privacy: Employing federated learning will enable training on distributed data while preserving privacy, crucial for sensitive smart city data [Mandal, 2024]. Key Research Questions: What are the trade-offs between privacy and model performance in federated learning? How can we address bias and fairness concerns? How can we adapt federated learning to real-time DT environments?

Ethical and Safety Considerations: Addressing ethical implications and ensuring safety and reliability involves developing mechanisms for accountability, transparency, bias detection, and robust error handling. Key Research Questions: How can we ensure accountability and transparency in decision-making? What are the potential biases, and how can we mitigate them? How can we establish ethical guidelines for cognitive DT systems?

Explainable AI (XAI) Integration: Integrating XAI techniques like rule-based explanation, attention visualization, provenance tracking, saliency maps, and counterfactual reasoning is crucial for transparency and trust. Key Research Questions: How can we effectively combine different XAI techniques for comprehensive explanations of the component workflow and processes with the CogTwin framework? How can we tailor explanations to different user groups? How can we evaluate the effectiveness of XAI in improving understanding and trust?

6 Conclusion

This paper introduced CogTwin, a novel hybrid cognitive architecture designed to significantly enhance DT autonomy and adaptability by integrating hybrid AI paradigms. Its key characteristics enable perception, reasoning, action, and learning within dynamic environments, with associated pseudocode providing a blueprint for development and preliminary evaluations suggesting the 50ms cognitive cycle's feasibility for real-time responsiveness. CogTwin represents a significant step towards achieving a unified theory of cognition within the DT domain, with the potential to transform complex system management. The immediate path forward prioritizes full software implementation and rigorous real-world validation in smart city pilot projects, followed by research into computational optimization and broader XAI integration as outlined in Section 5.

Realizing CogTwin's full potential, however, necessitates addressing critical future research avenues. Key among these are the empirical validation of the 50ms cycle across diverse hardware, the development of standardized CDT benchmarks crucial for assessing performance and adaptability, and the refinement of knowledge management processes, including DKR/DIKG conflict resolution and criteria for offline enrichment. Furthermore, the practical implementation and thorough evaluation of specific XAI techniques (Section 5.3) and empirically validating the Meta-Cognitive Layer's low-overhead adaptive control are vital next steps. Systematically addressing these areas will be crucial in translating the CogTwin framework into robust, deployable, and truly intelligent CDT systems, its reusable architecture enabling diverse solutions beyond specific applications, fostering transparency, trust, and effective human-CogTwin collaboration.

Acknowledgments

This publication has emanated from research conducted with the financial support of ML-Labs at Dublin City University under grant number 18/CRT/6183 and the Insight Research Ireland Centre for Data Analytics under grant number SFI/12/RC/2289_P2.

References

[Abburu *et al.*, 2020] Sailesh Abburu, Arne J Berre, Michael Jacoby, Dumitru Roman, Ljiljana Stojanovic, and Nenad Stojanovic. Cognitwin–hybrid and cognitive digital twins for the process industry. In *2020 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 1–8. IEEE, 2020.

[Anderson *et al.*, 2004] John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. An integrated theory of the mind. *Psychological review*, 111(4):1036, 2004.

[Anshakov and Gergely, 2010] Oleg M Anshakov and Tamás Gergely. *Cognitive reasoning: a formal approach*. Springer Science & Business Media, 2010.

[Ash, 2012] Robert B Ash. *Information theory*. Courier Corporation, 2012.

[Baheti and Gill, 2011] Radhakisan Baheti and Helen Gill. Cyber-physical systems. *The impact of control technology*, 12(1):161–166, 2011.

[Bettini *et al.*, 2010] Claudio Bettini, Oliver Brdiczka, Karen Henricksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, and Daniele Riboni. A survey of context modelling and reasoning techniques. *Pervasive and mobile computing*, 6(2):161–180, 2010.

[Bhuyan *et al.*, 2024] Bikram Pratim Bhuyan, Amar Ramdane-Cherif, Ravi Tomar, and TP Singh. Neuro-symbolic artificial intelligence: a survey. *Neural Computing and Applications*, pages 1–36, 2024.

[Buluç *et al.*, 2016] Aydın Buluç, Henning Meyerhenke, Ilya Safro, Peter Sanders, and Christian Schulz. *Recent advances in graph partitioning*. Springer, 2016.

[Byom and Mutlu, 2013] Lindsey J Byom and Bilge Mutlu. Theory of mind: Mechanisms, methods, and new directions. *Frontiers in human neuroscience*, 7:413, 2013.

[Calderita *et al.*, 2020] Luis V Calderita, Araceli Vega, Sergio Barroso-Ramírez, Pablo Bustos, and Pedro Núñez. Designing a cyber-physical system for ambient assisted living: A use-case analysis for social robot navigation in caregiving centers. *Sensors*, 20(14):4005, 2020.

[CogTwin, 2025a] CogTwin. CogTwin Data Flow: Perceptual Buffer to Internal KG Update. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinDataFlow.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025b] CogTwin. CogTwin Framework. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinFrame.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025c] CogTwin. Table: CogTwin Module Interactions. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinModules.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025d] CogTwin. Table: Comparison of CogTwin with Cognitive Architectures. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinCompArch.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025e] CogTwin. Table: Comparison of CogTwin with Representative Cognitive Digital Twin (CDT) Architectures. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinCompCDT.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025f] CogTwin. Use Case 1: Traffic Incident Management. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUC1.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025g] CogTwin. Use Case 2: Power Outage Response. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUC2.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025h] CogTwin. Use Case 3: Smart Home Automation. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUC3.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025i] CogTwin. Use Case 4: Medical Emergency Response. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUC4.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025j] CogTwin. Use Case: Cross-Domain Interaction. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUCX.md, 2025. Accessed: 2025-06-16.

[CogTwin, 2025k] CogTwin. Use Cases: Performance Evaluation. https://github.com/sukanyamandal/ProjectCogTwin/blob/main/materials/CogTwinUCPerf.md, 2025. Accessed: 2025-06-16.

[Cox, 2011] Michael T Cox. Metareasoning, monitoring, and self-explanation. 2011.

[Du *et al.*, 2020] Jing Du, Qi Zhu, Yangming Shi, Qi Wang, Yingzi Lin, and Daniel Zhao. Cognition digital twins for personalized information systems of smart cities: Proof of concept. *Journal of Management in Engineering*, 36(2):04019052, 2020.

[Eirinakis *et al.*, 2020] Pavlos Eirinakis, Kostas Kalaboukas, Stavros Lounis, Ioannis Mourtos, Jože M Rožanec, Nenad Stojanovic, and Georgios Zois. Enhancing cognition for digital twins. In *2020 IEEE International conference on engineering, technology and innovation (ICE/ITMC)*, pages 1–7. IEEE, 2020.

[Franklin *et al.*, 2013] Stan Franklin, Tamas Madl, Sidney D'mello, and Javier Snaider. Lida: A systems-level architecture for cognition, emotion, and learning. *IEEE Transactions on Autonomous Mental Development*, 6(1):19–41, 2013.

[Furbach *et al.*, 2019] Ulrich Furbach, Steffen Hölldobler, Marco Ragni, Claudia Schon, and Frieder Stolzenburg. Cognitive reasoning: A personal view. *KI-Künstliche Intelligenz*, 33:209–217, 2019.

[Harel and Pnueli, 1984] David Harel and Amir Pnueli. On the development of reactive systems. In *Logics and models of concurrent systems*, pages 477–498. Springer, 1984.

[Hribernik *et al.*, 2021] Karl Hribernik, Giacomo Cabri, Federica Mandreoli, and Gregoris Mentzas. Autonomous, context-aware, adaptive digital twins—state of the art and roadmap. *Computers in Industry*, 133:103508, 2021.

[Ilkou and Koutraki, 2020] Eleni Ilkou and Maria Koutraki. Symbolic vs sub-symbolic ai methods: Friends or enemies? In *CIKM (Workshops)*, volume 2699, 2020.

[Jinzhi *et al.*, 2022] Lu Jinzhi, Yang Zhaorui, Zheng Xiaochen, Wang Jian, and Kiritsis Dimitris. Exploring the concept of cognitive digital twin from model-based systems engineering perspective. *The International Journal of Advanced Manufacturing Technology*, 121(9):5835–5854, 2022.

[Jones *et al.*, 2020] David Jones, Chris Snider, Aydin Nassehi, Jason Yon, and Ben Hicks. Characterising the digital twin: A systematic literature review. *CIRP journal of manufacturing science and technology*, 29:36–52, 2020.

[Just and Carpenter, 1992] Marcel A Just and Patricia A Carpenter. A capacity theory of comprehension: individual differences in working memory. *Psychological review*, 99(1):122, 1992.

[Kahneman, 2011] Daniel Kahneman. Thinking, fast and slow. *Farrar, Straus and Giroux*, 2011.

[Khan *et al.*, 2023] Tarique Hasan Khan, Chiho Noh, and Soonhung Han. Correspondence measure: a review for the digital twin standardization. *The International Journal of Advanced Manufacturing Technology*, 128(5-6):1907–1927, 2023.

[Kieras and Meyer, 1997] Davis E Kieras and Davis E Meyer. An overview of the epic architecture for cognition and performance with application to human-computer interaction. *Human–Computer Interaction*, 12(4):391–438, 1997.

[Kobayashi and Alam, 2024] Kazuma Kobayashi and Syed Bahauddin Alam. Deep neural operator-driven real-time inference to enable digital twin solutions for nuclear energy systems. *Scientific reports*, 14(1):2101, 2024.

[Kunz *et al.*, 1984] John C Kunz, Thomas P Kehler, and Michael D Williams. Applications development using a hybrid artificial intelligence development system. *AI magazine*, 5(3):41–41, 1984.

[Laird, 2019] John E Laird. *The Soar cognitive architecture*. MIT press, 2019.

[Liu *et al.*, 2018] Yike Liu, Tara Safavi, Abhilash Dighe, and Danai Koutra. Graph summarization methods and applications: A survey. *ACM computing surveys (CSUR)*, 51(3):1–34, 2018.

[Liu *et al.*, 2025] Hui Liu, Wenyin Tao, Zhixin Hong, and Lianmin Shi. Real-time analysis of event response based on rt-thread. *IEEE Access*, 2025.

[Lu *et al.*, 2020] Jinzhi Lu, Xiaochen Zheng, Ali Gharaei, Kostas Kalaboukas, and Dimitris Kiritsis. Cognitive twins for supporting decision-makings of internet of things systems. In *Proceedings of 5th International Conference on the Industry 4.0 Model for Advanced Manufacturing: AMP 2020*, pages 105–115. Springer, 2020.

[Lv *et al.*, 2023] Jianhao Lv, Xinyu Li, Yicheng Sun, Yu Zheng, and Jinsong Bao. A bio-inspired lida cognitive-based digital twin architecture for unmanned maintenance of machine tools. *Robotics and Computer-Integrated Manufacturing*, 80:102489, 2023.

[Mandal and O'Connor, 2024] Sukanya Mandal and Noel E O'Connor. Llmasmmkg: Llm assisted synthetic multi-modal knowledge graph creation for smart city cognitive digital twins. In *Proceedings of the AAAI Symposium Series*, volume 4, pages 210–221, 2024.

[Mandal and O'Connor, 2024] Sukanya Mandal and Noel E O'Connor. Ontosmartdcu: A multi-agent microservice based automated, dynamic ontology and knowledge graph creation framework from real-time sensor data for the smart dcu digital twin. In *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, pages 784–789. IEEE, 2024.

[Mandal, 2024] Sukanya Mandal. A privacy preserving federated learning (ppfl) based cognitive digital twin (cdt) framework for smart cities. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23399–23400, 2024.

[Michelson, 2006] Brenda M Michelson. Event-driven architecture overview. *Patricia Seybold Group*, 2(12):10–1571, 2006.

[Newell, 1992] Allen Newell. Précis of unified theories of cognition. *Behavioral and Brain Sciences*, 15(3):425–437, 1992.

[Newell, 1994] Allen Newell. *Unified theories of cognition*. Harvard University Press, 1994.

[Park and Park, 2020] Ji Su Park and Jong Hyuk Park. Enhanced machine learning algorithms: deep learning, reinforcement learning, and q-learning. *Journal of Information Processing Systems*, 16(5):1001–1007, 2020.

[Parnafes and Disessa, 2004] Orit Parnafes and Andrea Disessa. Relations between types of reasoning and computational representations. *International Journal of Computers for Mathematical Learning*, 9:251–280, 2004.

[Prasad and others, 2012] TV Prasad et al. Hybrid systems for knowledge representation in artificial intelligence. *arXiv preprint arXiv:1211.2736*, 2012.

[Rogers, 2012] Yvonne Rogers. *HCI theory: classical, modern, and contemporary*, volume 14. Morgan & Claypool Publishers, 2012.

[Rosenbloom, 2013] Paul S Rosenbloom. The sigma cognitive architecture and system. *AISB Quarterly*, 136:4–13, 2013.

[S-Julián *et al.*, 2023] Raúl S-Julián, Ignacio Lacalle, Rafael Vaño, Fernando Boronat, and Carlos E Palau. Self-* capabilities of cloud-edge nodes: A research review. *Sensors*, 23(6):2931, 2023.

[Samsonovich, 2009] Alexei V Samsonovich. The constructor metacognitive architecture. In *2009 AAAI Fall Symposium Series*, 2009.

[Sha *et al.*, 2004] Lui Sha, Tarek Abdelzaher, Karl-Erik Årzén, Anton Cervin, Theodore Baker, Alan Burns, Giorgio Buttazzo, Marco Caccamo, John Lehoczky, and Aloysius K Mok. Real time scheduling theory: A historical perspective. *Real-time systems*, 28:101–155, 2004.

[Shahzad *et al.*,] Naveed Shahzad, William de Paula Ferreira, and Fernando Deschamps. Cognitive digital twins: A state-of-the-art review. *Available at SSRN 5085883*.

[Singh *et al.*, 2021] Maulshree Singh, Evert Fuenmayor, Eoin P Hinchy, Yuansong Qiao, Niall Murray, and Declan Devine. Digital twin: Origin to future. *Applied System Innovation*, 4(2):36, 2021.

[Smolensky, 1987] Paul Smolensky. Connectionist ai, symbolic ai, and the brain. *Artificial Intelligence Review*, 1(2):95–109, 1987.

[Stankovic and others, 1988] John A Stankovic et al. *Real-time computing systems: The next generation*. University of Massachusetts at Amherst. Computer and Information Science [COINS], 1988.

[Sun, 2006] Ron Sun. The clarion cognitive architecture: Extending cognitive modeling to social simulation. *Cognition and multi-agent interaction*, pages 79–99, 2006.

[Sycara, 1998] Katia P Sycara. Multiagent systems. *AI magazine*, 19(2):79–79, 1998.

[Thórisson and Helgasson, 2012] Kristinn Thórisson and Helgi Helgasson. Cognitive architectures and autonomy: A comparative review. *Journal of Artificial General Intelligence*, 3(2):1–30, 2012.

[Trafton *et al.*, 2013] J Gregory Trafton, Laura M Hiatt, Anthony M Harrison, Franklin P Tamborello, Sangeet S Khemlani, and Alan C Schultz. Act-r/e: An embodied cognitive architecture for human-robot interaction. *Journal of Human-Robot Interaction*, 2(1):30–55, 2013.

[Van der Hoek and Wooldridge, 2008] Wiebe Van der Hoek and Michael Wooldridge. Multi-agent systems. *Foundations of Artificial Intelligence*, 3:887–928, 2008.

[Vaswani, 2017] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[Zhao *et al.*, 2007] Peixiang Zhao, Jeffrey Xu Yu, and Philip S Yu. Graph indexing: tree+ delta¡= graph. In *Proceedings of the 33rd international conference on Very large data bases*, pages 938–949, 2007.

[Zheng *et al.*, 2022] Xiaochen Zheng, Jinzhi Lu, and Dimitris Kiritsis. The emergence of cognitive digital twin: vision, challenges and opportunities. *International Journal of Production Research*, 60(24):7610–7632, 2022.