FORMAL APPROACH TO DIGITAL TWIN SPECIFICATION

Mama Diakité
Mamadou Kaba Traoré

University of Bordeaux,
IMS UMR 5218
351 Cours de la Libération
33405 Talence, FRANCE
{mama.diakite, mamadou-kaba.traore}@u-bordeaux.fr

ABSTRACT

In a context of increasingly smart systems and ambient intelligence, the digital twin is a concept that is surfacing to help monitor, simulate, control and pilot such systems. As such, it is one of the new and disruptive trends in the area of Modeling and Simulation. However, despite numerous on-going research and development initiatives, there is a lack of formal approach to digital twin modeling and engineering. This work proposes a system-theoretic approach to achieve this goal. Such a framework allows formally and unambiguously specifying a digital twin, thus opening the way for both automated or semi-automated code synthesis, as well as symbolic manipulation to verification, validation, composition, reuse, and more.

Keywords: Digital Twin, formal specification, visual representation.

1 INTRODUCTION

The concept of "smart everything" is emerging with the ever-growing digitalization of society. Consequently, new types of systems are appearing, where data and virtual technologies occupy a prominent place. Such systems are so complex that their management requires model-based approaches. The Digital Twin (DT) concept has surfaced as such an approach and is landing in top strategic technology trends. It is based on the idea that a model which is used in different ways in place of a system of interest, is continuously synchronized with that system in order to reflect any real event happening to the system on the model, such that any management initiative can be assessed on this ever-updated artifact before transferring it to the system. Therefore, the model is more than a simple representation of the system, but a digital counterpart which is specifically bound to the system, rather than representing a family of systems of the same kind.

Due to the lack of rigorous formalization, there is no unique understanding of the concept of DT, as evidenced by the use of the word in very diverse professional contexts and applications (Smith et al., 2011; Stackpole, 2015; Johnson, 2016), and the flowering of definitions in the available literature (Glaessgen and Stargel, 2012; Lee et al., 2013; Rosen et al., 2015; Grieves and Vickers, 2016; Söderberg et al., 2017; Bolton et al., 2018; El Saddik, 2018; Tao et al., 2018). This results in debates about what a DT is, leading to ambiguity on the concept among stakeholders, and therefore on the solution to be developed.

Motivated by the need of a formal DT specification framework, this paper introduces a candidate formalism to achieve that aim. We first discuss in Section 2 the building blocks for a unified approach to the DT

ANNSIM '23, May 23-26, 2023, Mohawk College, ON, CANADA;

concept. Then, we propose in Section 3 a system-theoretic specification approach, capturing a Digital Twin as a dynamic system. We illustrate in Section 4 how this formalism applies. In Section 5, we discuss related works and the scope of our formalism. A conclusion and future work are given in Section 6.

2 DIGITAL TWIN CONCEPT

Various uses of the term Digital Twin in different communities raise the question of its formal definition. Nonetheless, we argue that the various Digital Twin viewpoints found in the literature fall under one common umbrella, as explained below.

2.1 Working Definitions

As illustrated by Figure 1, a model of a world of interest is a DT when and only when both are synchronized based on data tracked in the world of interest.

Definition 1. *A Twin of Interest (ToI) refers to a world of interest, seen from a system-theoretic perspective (i.e., a product, a service, or a process). As the world of interest can even be a software (therefore an immaterial/virtual entity), the term ToI is preferred to terms such as "physical twin" or "real twin".*

Definition 2. *A Digital Twin (DT) refers to a virtual model synchronized with a ToI. As a model, it is an abstraction that may reflect one or multiple perspectives (static, dynamic, functional, etc.) of the ToI.*

Figure 1: DT basic principle.

2.2 DT-ToI Synchronization

As illustrated by Figure 2, the synchronization between the ToI and the DT can be: (1) clock-based, ranging from high frequencies (real-time or near real-time synchronization), to low frequencies (cyclic synchronization) or (2) event-based, ranging from high predictability (conditional synchronization, i.e. synchronization triggered by the satisfaction of a condition), to low predictability (on-demand synchronization). Highly predictable ToIs require low synchronization frequency as there is a high confidence in the faithfulness of the model. On contrary, unpredictable ToIs call for a high synchronization frequency, as the model would quickly deviate from reality in the absence of updating data.

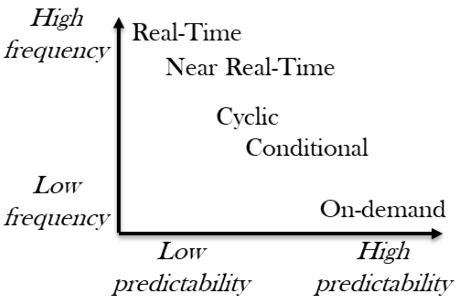

Figure 2: Characteristics of the DT-ToI twinning.

2.3 Closed versus Open DT-ToI Synchronization

While data sensed at the ToI side are directly sent to the DT side through a communication middleware, the decision derived from the DT side to be sent at the ToI side is not necessarily automated (as captured by the dotted line from DT to ToI in Figure 1). Figure 3 shows that there might be or not a human third party in the loop. Therefore, the twinning loop can either be closed (in which case, the DT not only monitors the ToI but also directly controls it) or open (in which case a third party handles the control decision).

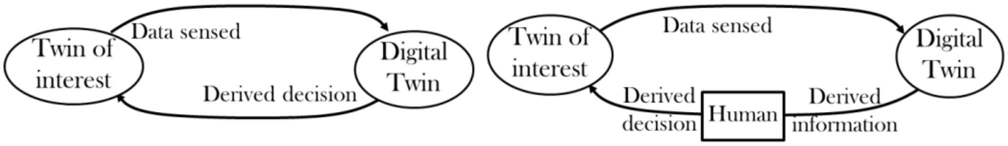

Figure 3: Closed and open twinning loops.

2.4 Role-Driven DT Functionalities

Table 1 summarizes roles and functionalities from existing DT viewpoints found in the literature (Piascik et al. 2012, Bailenson and Segovia 2010, Glaessgen and Stargel 2012, Reifsnider and Majumdar 2013, Rios et al. 2015, Bramlet et al. 2016, Grieves and Vickers 2016, Ben Miled and French 2017, Zhang et al. 2017, Negri, Fumagalli and Macchi 2017, Bolton et al. 2018, El Saddik 2018, Bauernhansl, Hartleif and Felix 2018, Park, Easwaran and Andalam 2019). Some of these references cross multiple viewpoints. The five major roles we identified are: (i) visual interface, (ii) diagnosis interface, (iii) prognostic interface, (iv) optimization interface, and (v) documentation interface. In each of the role, the DT can be used in open or closed loop for different goals, and therefore provide different functionalities:

- In visual monitoring, the DT is a replica based on current data sensed, which allow users visually monitoring the ToI indirectly and getting insights that they would not be able to get directly.
- In visual control, decisions are directly sent by the DT to the ToI, even if they are derived by a human operator. The DT serves as a proxy to the ToI.
- In data-based monitoring, not only current data sensed are used by the DT, but also historical data, thus allowing to detect trends and special situations. The DT serves as a dual of the ToI’s history.
- In automated control, the DT doesn’t need a human operator to derive decisions to be sent to the ToI. As such, it is the exact cyber counterpart of the ToI.
- In simulation-based forecasting, the DT allows what-if scenarios be explored, through discrete/continuous/3D-motion simulations, to predict the future of the ToI from data sensed.
- In real-time simulation-based decision making, faster than real what-if explorations are performed in order to take the best control decision for the ToI.
- In exploration-based design, the what-if scenarios exploration is done to select the best configuration for the redesign of some of the ToI’s components.
- In exploration-based decision making, the what-if scenarios exploration supports the decision making process that the DT uses to control the ToI.
- In data laking, the DT is the collection of all descriptions related to the ToI in all of its lifecycle phases (including from when the ToI is an idea to prototype and then to final entity).
- In information mining, the DT derives insights from the data lake to directly feed the ToI back, a general case where all the previously functionalities (closed loop cases) can be combined.

3 DIGITAL TWIN FORMAL SPECIFICATION AND VISUAL REPRESENTATION

The DT engineering requires capabilities for the DT model to self-update. This poses the theoretical problem of model inference from data collected on the system. It requires a formal framework, in which a system representation is coupled with learning methods to achieve automatic model updating. To formally specify a DT, we adopt a DEVS-based approach (Zeigler 1976), where the structure of the DT is defined as a black box with inputs and outputs interfacing it with its environment (through sensors and actuators). Inside the black box, the DT behavior is expressed as an automaton, describing the DT phases and phase-to-phase transitions (including triggering conditions), as well as how the DT gives feedback to its environment. Moreover, to ease modeling efforts, we also provide an associated visual notation. This alleviates potential complexity barriers in using the notation, paving the way to collaborative DT modeling.

Table 1: Role-driven DT functionalities.

Role	**Open loop functionality**	**Closed loop functionality**
Visual interface	Visual monitoring	Visual control
Diagnosis interface	Data-based monitoring	Automated control
Prognosis interface	Simulation-based forecasting	Real-Time decision-making
Optimization interface	Exploration-based design	Exploration-based decision-making
Documentation interface	Data laking	Information mining

3.1 Formal Specification

We specify a DT as a 7-uplets $M = \langle \Lambda, X, Y, \Phi, \Sigma, \Delta, \Delta^* \rangle$, where:

- Λ is the parameter set; it models all assumptions made on the context in which the DT model is used ($\forall \lambda \in \Lambda$, dom($\lambda$) is the set of all admissible values for λ).
- X is the input set; it models the influences received from the DT environment ($\forall x \in X$, dom(x) is the set of all admissible values for x).
- Y is the output set; it models the DT influences on the environment ($\forall y \in Y$, dom(y) is the set of all admissible values for y).
- Φ is the phase set; it models the stable discrete/continuous steps of the DT model.
- $\forall \varphi \in \Phi$, $\Phi_\varphi = \langle \pi_\varphi, \theta_\varphi, \tau_\varphi \rangle$ is the phase's definition, with:
  - → π_φ is a predicate on Σ (noted $\pi_\varphi \in \wp(\Sigma) \cup \{NIL\}$) giving the semantics of π in Σ
  - → θ_φ is a predicate on Σ modeling the activity performed during φ
  - → τ_φ is the time delay due to φ (i.e., expected lifetime of φ; $\tau_\varphi \in \mathbb{R}_+$)
- Σ is the semantic domain; this is the set of variables on which the phases are mapped ($\forall \sigma \in \Sigma$, dom($\sigma$) is the set of all admissible values for σ). Σ is defined by the variables provided by the sensors and the actuators through which the DT is twinned with the ToI.
- $\Delta = \{\Delta_{i?}^{j}, \Delta_{i}^{j!}, \Delta_{i?}^{j!}, i \in \Phi, j \in \Phi\}$ is the phase-to-phase transition set, with:
  - → $\Delta_{i?}^{j} = \langle \omega_{i?,j}, \varepsilon_{i?,j}, \theta_{i?,j} \rangle$ where $\omega_{i?,j} \in \wp(X)$ is the condition of input receipt,
    $\varepsilon_{i?,j} \in \wp(\Sigma) \cup \{NIL\}$ is the condition of transition, and
    $\theta_{i?,j} \in \wp(\Sigma) \cup \{NIL\}$ is the action performed in transition.
  - → $\Delta_{i}^{j!} = \langle \varepsilon_{i,j!}, \rho_{i,j!}, \theta_{i,j!} \rangle$ where $\varepsilon_{i,j!} \in \wp(\Sigma) \cup \{NIL\}$ is the condition of transition,
    $\rho_{i,j!} \in \wp(Y) \cup \{NIL\}$ is the output sending predicate, and
    $\theta_{i,j!} \in \wp(\Sigma) \cup \{NIL\}$ is the action performed in transition.
  - → $\Delta_{i?}^{j!} = \langle \omega_{i?,j!}, \varepsilon_{i?,j!}, \rho_{i?,j!}, \theta_{i?,j!} \rangle$ where $\omega_{i?,j!} \in \wp(X)$ is the condition of input receipt,
    $\varepsilon_{i?,j!} \in \wp(\Sigma) \cup \{NIL\}$ is the condition of transition,

$\rho_{i?,j!} \in \wp(Y) \cup \{NIL\}$ is the output predicate, and

$\theta_{i?,j!} \in \wp(\Sigma) \cup \{NIL\}$ is the action done in transition.

- $\Delta^* = \langle i^*, \theta^*, \tau^* \rangle$ is the initialization, giving the phase and action prior to the DT execution, where:
  - → i^* is the initial phase
  - → θ^* is the action done in initializing the model
  - → τ^* is the time already elapsed in the initial phase.

Note 1. *$\Delta_{i?}^{j}$, $\Delta_{i}^{j!}$, and $\Delta_{i?}^{j!}$ are respectively called external, internal, and confluent transitions (in the 1st case, such transition occurs when the delay of the current phase elapses; in the 2nd case, the transition is triggered by the receipt of an input; in the 3rd case, both transition conditions occur simultaneously).*

Note 2. *The predicate to specify the receipt of a value v in the input channel p reads p?v. The predicate to specify the sending of a value v through the output channel p reads p!v.*

Note 3. *The formal approach addresses both atomic and coupled models. For a coupled model, Σ specifies the model's components, and π_φ specifies for each phase how these components are coupled in that phase.*

3.2 Visual Representation

The main principles of our visual DT representation are shown by Figure 4:

- The whole model is a three-compartments box labeled with the name of the DT.
- Parameters are indicated by double-lined boxes, with for each of them the indication of the name and the domain of the corresponding parameter. Parameter boxes can be placed anywhere at the edge of the DT model box.
- Input and output channels (or ports) are represented by boxes with arrow inside and indication of their names and domains. Input and output boxes can also be placed anywhere at the edge of the DT model box. When the arrow is directed towards the box, this is an input, otherwise it's an output.
- The semantic domain of the DT model's phases is specified in the first compartment of the DT model box, i.e., all the variables belonging to this domain are indicated (name and domain).
- All predicates useful for the model specification can be specified in the second compartment. In practice, only functions to be called are specified as predicate schemas.
- The phases, transitions and initialization are specified in the last compartment.

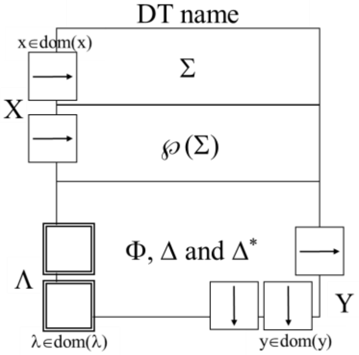

Figure 4: Visual DT representation.

The building blocks used in the last compartment of the visual DT representation are given in Figure 5:

- Figure 5.a shows a regular phase, a three-compartment circular node labeled by the phase name, and which compartments are respectively dedicated to the specification of the semantics of the phase, the activity performed, and the delay. Figure 5.b and Figure 5.c are special cases of phase, respectively for zero and infinite delay.

- Figure 5.d shows an internal transition, a straight-line arrow with indication of its related elements, while Figure 5.e shows an external transition, a dotted-line arrow, and Figure 5.f shows a confluent transition, a double-line arrow. Figure 5.g shows the same transitions in the case there is no guarding condition.
- Figure 5.h shows how the initialization process is visually specified, with indication of the initial elapsed time by the symbol of a clock.

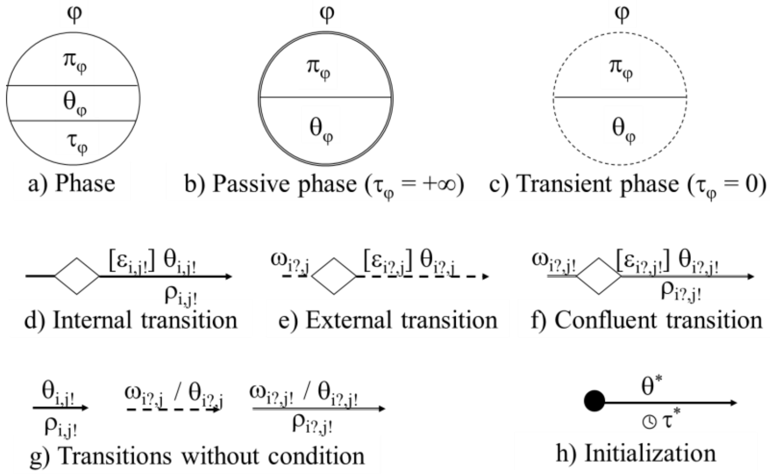

Figure 5: Primitive elements of the visual representation.

4 APPLICATION

Infrastructure and network operators, cities and manufacturers are faced with complex issues to ensure the viability of the service, maintain, operate and develop the mobility system, while integrating environmental, economic and societal impacts. Thus, opening or modifying a traffic axis, creating a public transport line on its own site, prohibiting car traffic in the city center, estimating the impact of the planned modifications on mobility and its uses, are all decisions that are heavy with consequences for the decision-maker both on his investments and on the consequences in terms of circulation. In this context, digital twins that integrate the multimodal impacts of mobility while taking into account the behavior of users, become a support for communities, transport operators and industrialists, in their diagnoses, the evaluation of projects and impact measurement.

4.1 Traffic of interest

In an effort to develop such a tool, the University of Bordeaux engaged in the engineering of a DT for the smart university campus. Sensors are deployed in many places in the campus area, and a prototype DT is being developed. Here we consider a small area of the whole project, focusing on one of the numerous campus' alternate traffic lanes, as illustrated by Figure 6. Two directions are considered: East to West, and West to East. A traffic regulator is alternatively allowing cars in the portion of the road located at the East side to move to the portion of the West side, and vice-versa (when one part sees go, the other part sees no-go, and vice-versa). Two sensors are deployed along the way, one sensing the car concentration in the West side, and the other in the East side. Both sensors are synchronized with the prototype DT.

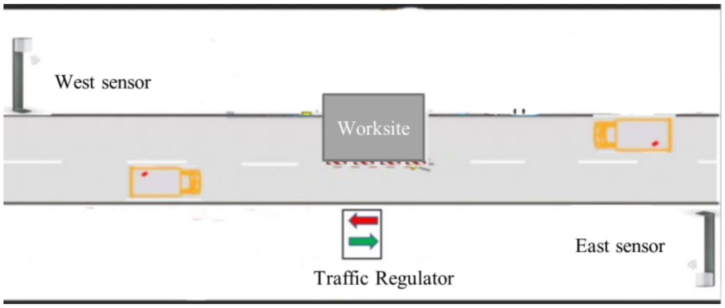

Figure 6: Traffic of interest.

4.2 Digital Twin Model Specification

Figure 7 gives the visual model of the ToI, indicating two roads (roadE and roadW, which belong to two different model classes RoadEast and RoadWest, as there is a slight difference in their respective behaviors), and two sources (sourceE and sourceW, which belong to the same car Generator class). The Traffic DT has two parameters (delayWE and delayEW), which respectively correspond to the durations of the green light (West to East, and East to West). The Generator class has a parameter indicating the rate of car generation, and an output port feed, which the cars are sent along. The RoadWest and RoadEst classes both have two parameters (speedWE and speedEW), which respectively give the average speed of cars in both directions (West to East, and East to West), two input ports (inWE and inEW) to receive cars moving respectively from West to East, and East to West, two output ports (outWE and outEW) to send cars respectively from West towards East, and from East towards West, and an input port sig to be notified of go or nogo.

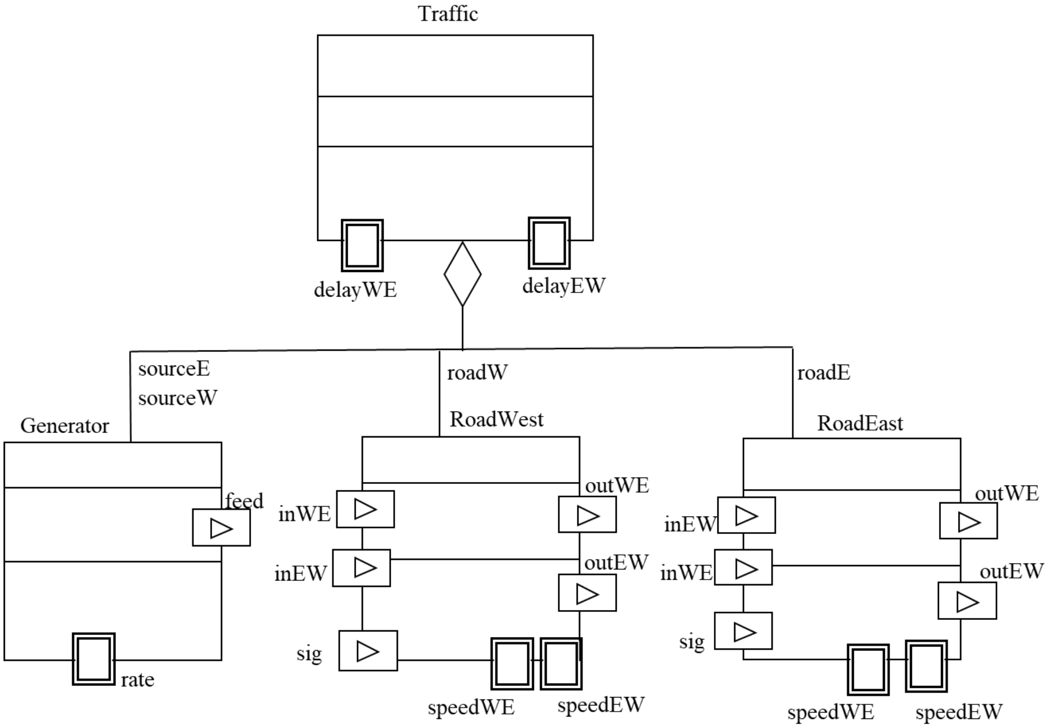

Figure 7: Traffic DT model.

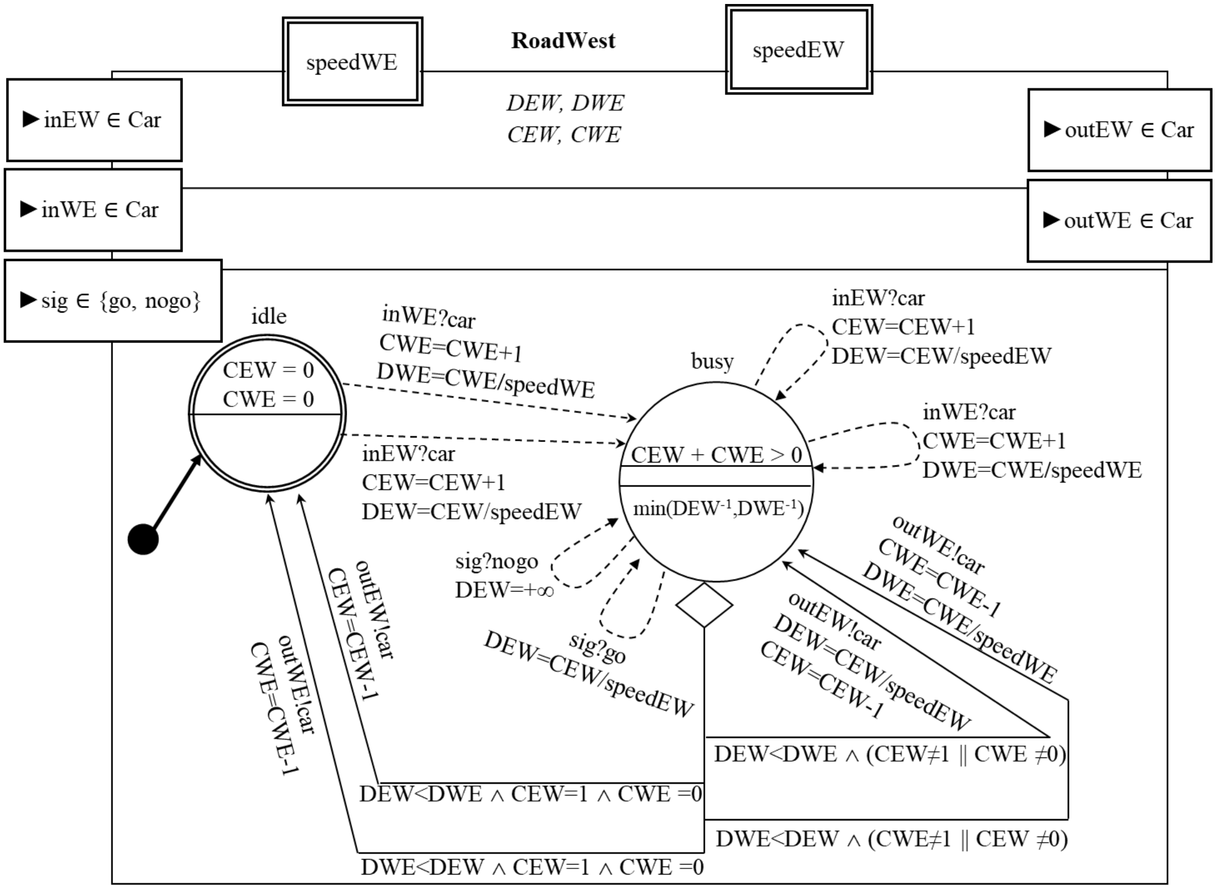

Figure 8: DT model for RoadWest.

Figure 8 shows the RoadWest model. Two phases are specified: idle when there is no car in none of the road portions, and busy otherwise. The semantic domain is given by the data that the deployed sensors can send to the DT, i.e., CEW (concentration of cars moving from East to West) and CWE (concentration of cars moving from West to East). Therefore, once synchronized, the DT is aware of the current situation of the ToI and thus can self-update accordingly. The corresponding formal specification is given by $DT_{RoadWest} = \langle \Lambda, X, Y, \Phi, \Sigma, \Delta, \Delta^{*} \rangle$, with:

- Λ = {speedWE, speedEW}
- X = {inEW, InWE, sig}
- Y = {outEW, outWE}
- Φ = {idle, busy}
  - → $\Phi_{idle} = \langle$CEW + CWE = 0, NIL, $+\infty\rangle$
  - → $\Phi_{busy} = \langle$CEW +CWE >0, NIL, min(1/DEW, 1/DWE)$\rangle$
- Σ = {DEW, DWE, CEW, CWE}
- $\Delta = \{{}_{1}\Delta_{busy?}^{busy}, {}_{2}\Delta_{busy?}^{busy}, {}_{3}\Delta_{busy?}^{busy}, {}_{4}\Delta_{busy?}^{busy}, {}_{1}\Delta_{idle?}^{idle}, {}_{2}\Delta_{idle?}^{idle}, {}_{1}\Delta_{busy}^{busy!}, {}_{2}\Delta_{busy}^{busy!}, {}_{1}\Delta_{busy}^{idle!}, {}_{2}\Delta_{busy}^{idle!}\}$ where a k index is used to differentiate transitions of same type with same adjacent phases
  - → ${}_{1}\Delta_{busy?}^{busy} = \langle$inEW?car, CEW = CEW+1 $\wedge$ DEW = CEW/speedEW$\rangle$
  - → ${}_{2}\Delta_{busy?}^{busy} = \langle$inWE?car, CWE = CWE+1 $\wedge$ DWE = CWE/speedWE$\rangle$
  - → ${}_{3}\Delta_{busy?}^{busy} = \langle$sig?go, DEW = CEW/speedEW$\rangle$
  - → ${}_{4}\Delta_{busy?}^{busy} = \langle$sig?nogo, DEW = $+\infty\rangle$
  - → ${}_{1}\Delta_{idle?}^{idle} = \langle$inEW?car, CEW=CEW+1 $\wedge$ DEW=CEW/speedEW$\rangle$
  - → ${}_{2}\Delta_{idle?}^{idle} = \langle$inWE?car, CWE = CWE+1 $\wedge$ DWE = CWE/speedWE$\rangle$

  - → ${}_{1}\Delta_{busy}^{busy!}$= ⟨DEW<DWE ∧ (CEW≠1 ∥ CWE ≠0), outEW!car, CEW= CEW-1 ∧ DEW= CEW/speedEW⟩
  - → ${}_{2}\Delta_{busy}^{busy!}$= ⟨DEW>DWE ∧ (CEW≠0 ∥ CWE ≠1), outWE!ar, CWE= CWE-1 ∧ DWE= CWE/speedWE⟩
  - → ${}_{1}\Delta_{busy}^{idle!}$= ⟨DEW<DWE ∧ CEW=1 ∧ CWE =0, outEW!car, CEW= CEW-1⟩
  - → ${}_{2}\Delta_{busy}^{idle!}$= ⟨DWE<DEW, CEW=0 ∧ CWE =1, outWE!car, CWE= CWE-1⟩
- Δ* = ⟨idle, CEW=0 ∧ CWE=0, 0⟩

The model of RoadEast is completely similar, except for the external transition triggered by the sig input port: when nogo is received, then DWE = +∞, and when go is received, then DWE = CWE/speedWE. Notice that DWE (respectively DEW) indicates the throughput of cars moving from West to East (respectively from East to West).

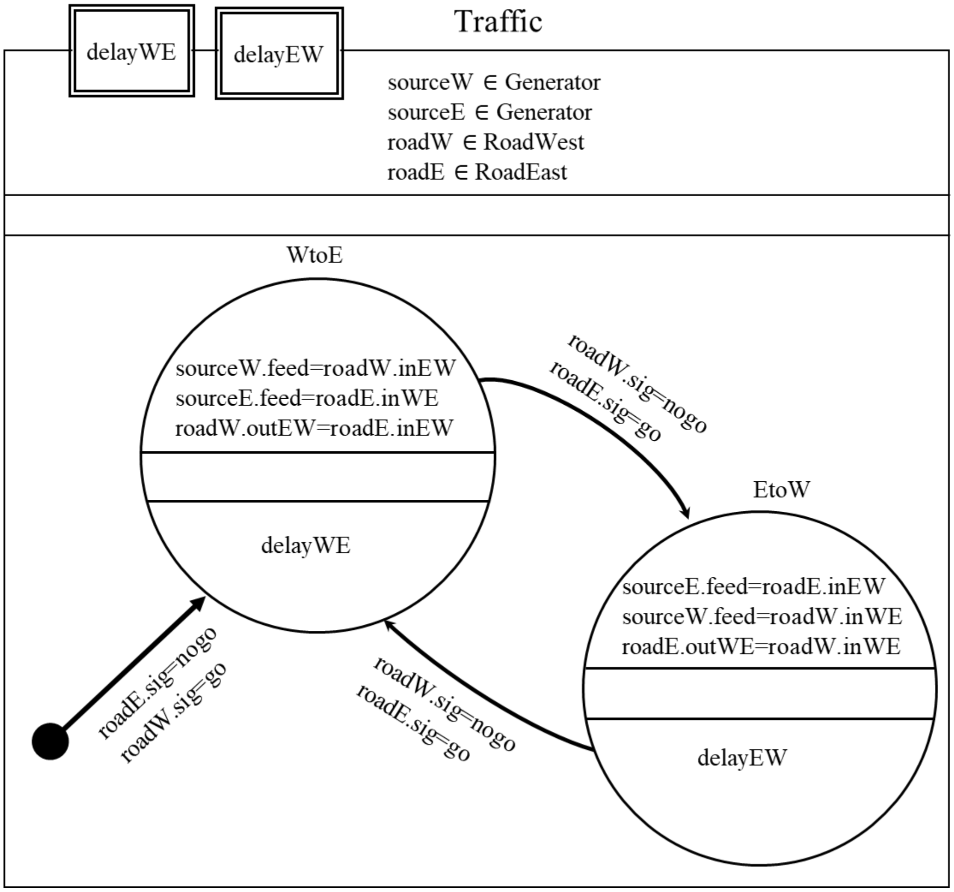

Figure 9: Traffic DT's behavioral rules.

Figure 10 gives the detailed description of the overall Traffic DT's behavioral rules. The corresponding formal specification is given by $DT_{Traffic}$ = ⟨Λ, X, Y, Φ, Σ, Δ, Δ^{*}⟩, with:

- Λ = {delayWE, delayEW}
- X = Y = ∅
- Φ = {WtoE, EtoW}
  - → Φ_{WtoE} = ⟨sourceW.feed=roadW.inEW ∧ sourceE.feed=roadE.inWE ∧ roadW.outEW=roadE.inEW, NIL, delayWE⟩

  - → $\Phi_{\text{EtoW}} = \langle$sourceE.feed=roadE.inEW $\wedge$ sourceW.feed=roadW.inWE $\wedge$ roadE.outWE=roadW.inWE, NIL, delayEW$\rangle$
- $\Sigma = \{$sourceW, sourceE, roadW, roadE$\}$
- $\Delta = \{\Delta_{\text{WtoE}}^{\text{EtoW!}}, \Delta_{\text{EtoW}}^{\text{WtoE!}}\}$
  - → $\Delta_{\text{WtoE}}^{\text{EtoW!}} = \langle$NIL, NIL, roadW.sig = nogo $\wedge$ roadE.sig = go$\rangle$
  - → $\Delta_{\text{EtoW}}^{\text{WtoE!}} = \langle$NIL, NIL, roadE.sig = nogo $\wedge$ roadW.sig = go$\rangle$
- $\Delta^* = \langle$WtoE, roadE.sig=nogo$\wedge$roadW.sig=go, 0$\rangle$

5 RELATED WORKS

Some notable efforts have been done in trying to capture the diversity of DT approaches. Some of them are surveys of the DT concepts in the literature, with in some cases an attempt to provide formal definitions or at least characterizations. Some other are reference models for DT in specific domains. Negri, Fumagalli and Macchi (2017) analyzed the definitions of the DT concept in scientific literature, retracing it from the initial conceptualization to most recent interpretations in Industry 4.0 and smart manufacturing research. The authors also proposed a definition of DT for Industry 4.0 manufacturing, in the context of the MAYA project, as a contribution to the research discussion about DT concept. Jones et al. (2020) provided a characterization of DT, identification of gaps in knowledge, and required areas of future research, based on a review and a thematic analysis of almost 100 publications on the topic. Monsone, Mercier-Laurent and János (2019) presented an overview of the role of DTs in transforming industrial ecosystems and discuss also the environmental impact. Kritzinger et al. (2018) provided a DT-related literature review with a categorization of contributions in terms of integration levels, focused areas, and technologies used. Our work is unique in that it proposes a formal framework adopting a system-theoretic approach. We also provide a visual notation that concretizes the abstract syntax.

6 CONCLUSION

This work proposes a framework to support Digital Twin formal specification. The benefits of formal specification are well-recognized. Firstly, it disambiguates the concept targeted. Secondly, it allows defining systematic methods for executable model generation. Thirdly, it opens the way to symbolic manipulation towards formally checking the inner consistency of the specification, as well as its validity against some requirements. What makes this formal approach suitable for DT specification is the notion of semantic domain, which maps the variables measured by sensors deployed in the field onto the possible phases of the DT. The necessary synchronization between the DT and its real counterpart is then straightforward. As regards to the DT roles (presented in Section 2.4), the formal approach proposed is adapted to the Prognosis interface roles (Simulation-based forecasting, and Real-Time decision-making). Consequently, it can also serve the Optimization interface roles (Exploration-based design, and Exploration-based decision-making).

Our next research efforts focus on building on top of the formal approach introduced, a method to combine simulation with deep/reinforcement learning techniques to achieve automatic model update based on data collected from real-world sensors and data sources. This is a requirement for the advent of a full-fledged DT technology. Also, there is a need to discuss theoretical and experimental aspects related to DT-ToI synchronization, including closed and open configurations. More specifically, the synchronization of components on different hierarchical levels at different rates needs to be supported in order to make such an approach really useable in an industrial context. Similarly, a supporting software tool for graphical modeling is of all importance to allow automatic generation of DT formal specifications, as well as symbolic reasoning about its correctness and the derivation of advanced insights. Such a tool will also allow to test the specification in real time or compare the phase transitions with the real system.

REFERENCES

Bailenson, J.N., and K. Y. Segovia. 2010. Virtual Doppelgangers: Psychological Effects of Avatars who Ignore their Owners. In *Online Worlds: Convergence of the Real and the Virtual*, ed. W. S. Bainbridge (London: Springer), pp. 175–186.

Bauernhansl, T., S. Hartleif, and T. Felix. 2018. "The Digital Shadow of Production – A Concept for the Effective and Efficient Information Supply in Dynamic Industrial Environments". *51st CIRP Conference on Manufacturing Systems*, pp. 69-74. Stockholm, Sweden.

Ben Miled, Z., and M. O. Frenc. 2017. "Towards a Reasoning Framework for Digital Clones using the Digital Thread". *55th AIAA Aerospace Sciences Meeting*, pp. 0873. Grapevine, TX (USA).

Bolton, R.N., J. R. McColl-Kennedy, L. Cheung, A. Gallan, C. Orsingher, L. Witell, and M. Zaki. 2018. "Customer experience challenges: Bringing together digital, physical and social realms". *J. of Service Management* vol. 29 (5), pp. 776–808.

Bramlet, M., K. Wang, A. Clemons, N. C. Speidel, S. M. Lavalle, and T. Kesavadas. 2016. "Virtual Reality Visualization of Patient Specific Heart Model". *J. of Cardiovascular Magnetic Resonance*, 18: T13.

El Saddik, A. 2018. "Digital Twins: The Convergence of Multimedia Technologies". *IEEE MultiMedia* vol. 25(2), pp. 87–92.

Glaessgen, E., and D. Stargel. 2012. "The Digital Twin Paradigm for Future NASA and US Air Force Vehicles". *53rd AIAA/ASME/ASCE/AHS/ASC*, pp. 1818. Honolulu, HI.

Grieves, M., and J. Vickers. 2016. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, in *Trans-Disciplinary Perspectives on System Complexity*, F.-J. Kahlen, S. Flumerfelt, and A. Alves, Ed. Springer: Switzerland. pp. 85-114.

Johnson, G. 2016. "The Digital Twin: Revolutionizing the Product and the Process". Process Online, http://processonline.com.au/content/software-it/article/the-digital-twin-revolutionising-the-product-and-the-process-733095522.

Jones, D., C. Snider, A. Nassehi, J. Yon, and B. Hicks. 2020. "Characterising the Digital Twin: a Systematic Literature Review". *CIRP J. Manuf. Sci. Technol*, n° 544.

Kritzinger, W., M. Karner, G. Traar, J. Henjes, and W. Sihn. 2018. "Digital Twin in manufacturing: A Categorical Literature Review and Classification". *IFAC-PapersOnLine* vol. 51(11), pp. 1016-1022, https://doi.org/10.1016/j.ifacol.2018.08.474.

Lee, J., E. Lapira, B. Bagheri, and H. Kao. 2013. Recent Advances and Trends in Predictive Manufacturing Systems in Big Data Environment. *Manufacturing letters* vol. 1, pp. 38-41.

Monsone, C., E. Mercier-Laurent, and J. János. 2019. "The Overview of Digital Twins in Industry 4.0: Managing the Whole Ecosystem". *11th International Conference on Knowledge Management and Information Systems*, p. 271-276. Vienna, France.

Negri, E., L. Fumagalli, and M. Macchi. 2017. "A Review of the Roles of Digital Twin in CPS-based Production Systems". In *Procedia Manufacturing*, vol. 11, pp. 939–948.

Park, H., A. Easwaran, and S. Andalam. 2019. Challenges in Digital Twin Development for Cyber-Physical Production Systems. In: *Chamberlain R., Taha W., Törngren M. (eds) Cyber Physical Systems. Model-Based Design. LN in CS*, vol 11615. Springer, Cham.

Piascik, R., J. Vickers, D. Lowry, S. Scotti, J. Stewart, and A. Calomino. 2012. "Technology Area 12: Materials, Structures, Mechanical Systems, and Manufacturing Road Map". NASA Office of Chief Technologist.

Reifsnider, K., and P. Majumdar. 2013. "Multiphysics Stimulated Simulation Digital Twin Methods for Fleet Management". *54th AIAA/ASME/ASCE/AHS/ASC*, pp. 1578. Boston, MA (USA).

Rios, J., J. C. Hernández, M. Oliva, and F. Mas. 2015. “Product Avatar as Digital Counterpart of a Physical Individual Product: Literature Review and Implications in an Aircraft System”. In *Proc. of ISPE Conference on Concurrent Engineering*, pp. 657–666. Delft, Netherland.

Rosen, R., G. von Wichert, G. Lo, and K. D. Bettenhausen. 2015. “About The Importance of Autonomy and Digital Twins for the Future of Manufacturing”. *IFAC-PapersOnLine*, vol. 48(3), pp. 567-572.

Schroeder, G., C. Steinmetz, C. E. Pereira, I. Muller, N. Garcia, D. Espindola, and R. Rodrigues. 2016. “Visualizing the Digital Twin using Web Services and Augmented Reality”. *IEEE 14th Int. Conf. on Ind. Informatics*, pp. 522-527. Poitiers, France.

Smith, N., A. De Vecchi, M. McCormick, D. Nordsletten, O. Camara, A. F. Frangi, and R. Rezavi. 2011. “EuHeart: Personalized and Integrated Cardiac Care using Patient-specific Cardiovascular Modelling”. *Int. Focus* vol. 1, pp. 349-364.

Stackpole, B. 2015. “Digital Twins Land: a Role in Product Design. Desktop Engineering”, Peerless Media LLC, http://www.deskeng.com/de/digital-twins-land-a-role-in-product-design/

Tao, F., F. Sui, A. Liu, Q. Qi, M. Zhang, B. Song, Z. Guo, S. C.-Y. Lu, and A. Y. C. Nee. 2018. “Digital Twin-driven Product Design Framework”. *Int. Journal of Production Research*, pp. 1-19.

Zeigler, B. P. 1976. *Theory of Modeling and Simulation*. John Wiley.

Zhang, H., Q. Liu, X. Chen, D. Zhang, and J. Leng. 2017. “A Digital Twin-based Approach for Designing and Multi-objective Optimization of Hollow Glass Production Line”. *IEEE Access*, n° 5, pp. 26901-26911.

AUTHOR BIOGRAPHIES

MAMA DIAKITE is a PhD student at the University of Bordeaux (France). She holds an MSc in Computer Science from Université des Sciences, Techniques et Technologies de Bamako (USTTB) in Mali. Her research interests lie in hybrid modeling and simulation to digital twin engineering. Her email address is mama.diakite@u-bordeaux.fr.

MAMADOU K. TRAORE is full Professor at the University of Bordeaux (France). He holds a PhD in Computer Science from Université Clermont Auvergne in France. His current research is in hybrid M&S for smart systems engineering. His email address is mamadou-kaba.traore@u-bordeaux.fr.