A continuous-time model-based approach for activity recognition in pervasive environments

Marco Biagi, Laura Carnevali, Marco Paolieri, Fulvio Patara, and Enrico Vicario, IEEE member

Abstract—We present a model-based approach to Activity Recognition (AR) in Ambient Assisted Living (AAL). The approach leverages an a-priori stochastic model termed Continuous Time Hidden Semi-Markov Model (CT-HSMM), capturing the continuous-time durations of activities and inter-event times. The model is enhanced according to the observed statistics, associating the events with an occurrence probability, and the sojourn time and the inter-event time in each activity with a continuous-time Probability Density Function (PDF), allowing effective fitting of observed durations through non-Markovian distributions. The model is updated at run-time according to a sequence of time-stamped observations, exploiting the method of stochastic state classes to perform transient analysis and derive a measure of likelihood that an activity is currently performed. The approach supports both online AR, predicting the activity performed at time $t$ using only the events observed until that time, and offline AR, applying a Forward-Backward procedure that exploits all the events observed before and after time $t$. The approach is experimented on a real data set of the literature, providing performance measures that can be compared with those of offline Hidden Markov Models (HMMs) and offline Hidden Semi-Markov Models (HSMMs).

Index Terms—Activity Recognition (AR), Ambient Assisted Living (AAL), continuous-time stochastic model, Semi-Markov Process (SMP), transient analysis, model@run.time.

I. INTRODUCTION

SMART homes support monitoring of high level human activities through typed and timestamped low-level observations collected by physical sensors, enabling services for healthcare, safety, security, and comfort, with notable applications in Ambient Assisted Living (AAL) [1]. To this end, Activity Recognition (AR) comprises a key challenge which has been widely addressed in pervasive computing systems [2] through data-driven and knowledge-based models [3], with various time abstractions enabling different exploitation of information carried by observed durations.

Data-driven approaches for AR have largely developed on Hidden Markov Models (HMMs) [4] and Conditional Random Fields (CRFs) [5]. In the HMM approach, activities are represented as hidden states of a discrete time Markov chain and an observation depending only on the current visited state is generated at each transition. Stochastic parameters of the HMM can be derived from a ground truth, through supervised learning or in closed form when model states are identified with predefined activities, and a maximum likelihood sequence of hidden activities corresponding to an observation sequence can be identified through efficient algorithms. CRFs relax the assumption of independent observations by directly modeling the conditional probability of a sequence of activities given a sequence of observations. HMMs and CRFs are applied and compared in [6] in the recognition of predefined Activities of Daily Living (ADLs) from sensor readings acquired in a smart home setting. In [7], activities are discovered by clustering emerging patterns through an unsupervised approach, an HMM is constructed for each cluster, and classifications are combined through a voting mechanism.

The sojourn duration in each hidden state of an HMM is implicitly associated with a geometric distribution, which is memoryless and fully identified by a single parameter, thus limiting exploitation of timing information that may be significant in various contexts [8]. The duration of activities is considered in [9] using Hidden Semi-Markov Models (HSMM) [10] and Semi-Markov Conditional Random Fields (SMCRFs) [11], which extend HMMs and CRFs, respectively, by explicitly modeling the distribution of the time segment during which the hidden state sojourns in a logical location. Interval State Hidden Semi Markov Models (IS-HSMM) [12] extend HSMMs with additional states representing no-observation intervals between consecutive events. Similarly, Interval Length Probability Hidden Semi-Markov Models (ILP-HSMMs) model the discrete-time distribution of such intervals [12]. In [13], HSMMs are extended to consider long-range dependencies between observations.

As a common trait, HMMs, CRFs, and their semi-Markov extensions all develop on a discrete abstraction of time, which in AR is usually realized by sampling the continuous underlying process at equidistant time points through the assumption of a time-tick. A few models were proposed to preserve a continuous time representation so as to support model compositionality, or to avoid the trade-off between accuracy and complexity that affects discretization, or more generally to fit the native characteristics of the observation process. Continuous-time HMMs were proposed in [14] in the context of performance evaluation to enable data-driven identification of a Continuous Time Markov Chain (CTMC). The problem can be reduced to the identification of a (discrete time) HMM which is then enhanced with continuous sojourn times, which however are necessarily distributed as memoryless exponential random variables. In Hidden non-Markovian Models (Hn-
MMs) [15], sojourn times in hidden states are associated with non-Markovian (i.e. non exponential) continuous distributions under the assumption that every change of the hidden state is made observable by the emission of an event, and approximate transient probabilities are derived by discretization through the Proxel simulation method [16] and used as a measure of likelihood for the identification of the hidden state. Non-Markovian sojourn times are considered also in Generalized Hidden Semi Markov Models (GHSMMs) [17], for which efficient algorithms are provided for training and inference under the assumption of a one-to-one mapping between hidden states and observation symbols.

Knowledge-based approaches reduce dependency on the volume of training data through pre-defined models capturing a-priori knowledge about hidden states. Online AR is addressed in [18], using ontologies for activity modeling and semantic reasoning for activity inferencing, supporting AR at coarse-grained and fine-grained levels [19] and adaptation of ADLs models over time [20]. In [21], a top-level ontology links knowledge implicit in elementary information to higher-level activity descriptions. In process mining [22], knowledge is extracted from event logs and used to improve processes [23]. Good results have been achieved in reconstruction of administrative workflows [24], while applicability appears more difficult for (less structured) healthcare pathways [25]. Techniques for process enhancement improve models with information extracted from observations, e.g., they enrich untimed models with continuous time stochastic parameters [26].

In this paper, we largely extend preliminary results of [27], presenting a novel approach to AR that exploits the discrimination capability of continuous-time durations of activities and inter-occurrence times of observed events. The approach relies on a novel generative stochastic model termed Continuous Time Hidden Semi-Markov Model (CT-HSMM), which captures the statistics of sojourn time in hidden activities and of the frequency of typed events observed in each of them, with notable extensions with respect to HnMMs and GHSMMs. In particular, the model accepts a randomized number of events observed during the sojourn in each activity and represents durations with non-Markovian distributions, making the likelihood of activities independent on the time elapsed between observations. Transient probabilities of future behavior until the next events conditioned on the actual sequence of past observations are efficiently computed at runtime [28] using the method of stochastic state classes [29], [30], and they are used as a measure of likelihood for the identification of the current ongoing hidden activity. The proposed approach enables both online and offline classification, yielding performance measures that can be compared with those of state-of-the-art offline approaches on a real data set of the AAL literature [6].

In the following: we formulate the problem and present the computed statistics (Section II); we define CT-HSMMs and perform model enhancement extending [31] to fit sample data (Section III); we present a solution technique for online and offline classification, tailoring the method of stochastic state classes [29], [30] to derive the likelihood of activities (Section IV); we discuss experimental results, (Section V), and, we draw our conclusions (Section VI). Technical details on stochastic state classes and on the proposed fitting approach are reported in the Supplemental Material with theorem proofs.

II. PROBLEM FORMULATION: AR IN AAL

A. An abstraction for activities and events

AR in AAL is a problem of partial observability where high-level activities are inferred from low-level events about human-object interactions. Events are typically captured by sensors (e.g., reed switches [6]) as typed and time-stamped observations (e.g., Opening fridge at 4:20pm). Similarly, activities performed by a human subject are characterized by a type in the index of Activities of Daily Living (ADLs) [32] (e.g., Preparing dinner) and by the time period (identified by a starting and an ending time-stamp) during which a stream of events is generated while accomplishing the activity goal. A special type of activity termed Idling identifies the time intervals during which no specified activity is performed.

The process of performed activities and observed events is uncertain, with different activities sharing the same event type (e.g., Opening/Closing fridge is a common event of Preparing dinner and Preparing a beverage) and different event types occurring in the same activity (e.g., Opening/Closing fridge and Starting/Ending dishwasher for Preparing dinner). Moreover, human activities are typically performed in complex and loosely structured manners, with limited repetitiveness. In this conceptual framework, AR consists in evaluating a measure of likelihood that an activity is currently performed, starting from a historical data set containing a stream of typed and time-stamped events with ground truth data about tagged activities.

B. A statistical characterization of AR data sets

The volume of available data is a key factor in the selection of statistics that can be affordably exploited. The proposed approach addresses recognition of sequential activities from events generated by switch sensors, leveraging five statistics on the frequency of events, the continuous duration of each activity, and the continuous time between consecutive events. As shown in Fig. 1, let: \( E = \{ e_1, \ldots, e_N \} \) and \( A = \{ a_1, \ldots, a_K \} \) be the sets of event types and activity types, respectively; \( a_n \) be the Idling type; \( Ev = \{e_{vn} = (e_{X(n)}, t_{vn})\}_{n=1}^{N} \) be the list of observed events in dual change-point representation (distinguishing activation and de-activation events), where \( e_{X(n)} \) and \( t_{vn} \) are the type and time of the \( n \)-th event, respectively; and, \( Act = \{act_{m} = (a_{y(m)}, t_{m}^{start}, t_{m}^{end})\}_{m=1}^{M} \) be the list of tagged activities, where \( a_{y(m)} \) is the type of the \( m \)-th activity, and \( t_{m}^{start} \) and \( t_{m}^{end} \) are its start and end times, respectively.

Fig. 1. A stream of observed events with sequential annotated activities. Circle, square and diamond pins represent punctual events with different types.
1) $F_a(a_k)$ is the duration histogram of activity type $a_k$; if $a_k \neq a_s$, duration is the time $\delta_m$ from the first to the last event of activity $act_m$ of type $a_k$; if $a_k = a_s$, it is the time $\delta_{m-1,m}$ from the last event of activity $act_{m-1}$ to the first event of the consecutive activity $act_m$. Tagged durations are not considered to avoid dependency on annotated data sets (not always available) and tagging errors (start and end times are often delayed/anticipated).

2) $F_I(a_k)$ is the inter-event time histogram of activity type $a_k$: if $a_k \neq a_s$, inter-event time is the time $\delta_{n+1,n-1}$ between consecutive events $ev_n$ and $ev_{n+1}$ in activity of type $a_k$; if $a_k = a_s$, it is the time between consecutive events $ev_n$, $ev_{n+1}$ s.t. $ev_n$ occurs in $Idling$, and $ev_{n+1}$ occurs in $Idling$ or the first event of a tagged activity (e.g., $[ev_{n-2}, ev_{n-1}]$ and $[ev_{n-1}, ev_n]$ in Fig. 1).

3) $P_e(e_h, a_k) =$ the frequency of an event of type $e_h$ in an activity of type $a_k$, i.e., $P_e(e_h, a_k) := \frac{E_e(e_h, a_k)}{E(a_k)}$, where $E(e_h, a_k)$ is the number of events of type $e_h$ observed in activities of type $a_k$ and $E(a_k)$ is the number of any type emitted in activities of type $a_k$.

4) $P_y(e_h)$ is the frequency of an event of type $e_h$ to occur during $Idling$ or as the starting event of a tagged activity, i.e., $P_y(e_h) := \frac{E_{y}(e_h)}{E_{y}(a_k)}$, where $E_{y}(e_h)$ is the number of events of type $e_h$ occurring during $Idling$ or as the starting event of a tagged activity, and $E_{y}(a_k)$ is the number of events of any type occurring during $Idling$ or as the starting event of a tagged activity.

5) $P_z(e_h, a_k)$ is the frequency of event of type $e_h$ to be the starting event of activity of type $a_k$, i.e., $P_z(e_h, a_k) := \frac{E_z(e_h, a_k)}{E(a_k)}$, where $E_z(e_h, a_k)$ is the number of events of type $e_h$ that occur as the first event of activity of type $a_k$.

### III. Continuous Time Hidden Semi-Markov Model

Information carried by observation data is captured into a generative model which we term Continuous Time Hidden Semi Markov Model (CT-HSMM). The state of the model consists of the current activity (either tagged or $Idling$), the remaining sojourn time, and the remaining time to the next observable event. An observable event may maintain the model in the current activity or may bring the model from $Idling$ to an annotated activity; conversely, expiration of the remaining sojourn time in an annotated activity and in $Idling$ is an unobservable event that brings the model to $Idling$ and to a deadlock location, respectively. The model is enhanced through the computed statistics, associating observable events with a probability, and the sojourn time and inter-event time in each activity with a Probability Density Function (PDF).

#### A. Model definition

**Syntax.** A CT-HSMM is a tuple $M := (A, a_s, E, P_x, P_y, P_z, T_0, T_a, F_d, F_i) \uparrow$ where:

- $A = \{a_1, \ldots, a_K\}$ is the set of annotated activities, $a_s$ is the $Idling$ activity, and $A_s = A \cup \{a_s\}$.

1To simplify presentation, in the CT-HSMM model we refer to activity types and event types as activities and events, respectively.

- $E = \{e_1, \ldots, e_H\}$ is the set of observable events.
- $P_x : E \times A \rightarrow [0, 1]$ is the probability that $e \in E$ occurs in $a \in A$, i.e., $P_x(e, a) := P[\epsilon(t) = e | \alpha(t) = \alpha_{(t_pr)}] = a]$, $\epsilon(t)$ being the event observed at $t$, $\alpha(t)$ the activity performed at $t$, and $t_{pr}$ the last observation time before $t$.
- $P_y : E \rightarrow [0, 1]$ is the probability that $e \in E$ is the first event in a tagged activity or occurs in $Idling$, i.e., $P_y(e) := P[\epsilon(t) = e | \alpha(t) = a_s \lor \epsilon(t) \text{ 1st  ev. } a(t) \in A]$.
- $P_z : E \times A \rightarrow [0, 1]$ is the probability that an event is the first one in $a \in A$ given that it is $e \in E$ and occurs in $Idling$ or as the first event of a tagged activity, i.e., $P_z(e, a) := P[\epsilon(t) \text{ 1st  ev. } a(t) = a | \epsilon(t) = e \land (\alpha_{(t)} = a_s \lor \epsilon(t) \text{ 1st  ev. } a(t) \in A)]$.
- $T_o \subseteq A \times E \times A$, is the set of observable transitions from an activity ($Idling$ or tagged) to itself and from $Idling$ to a tagged activity, i.e., $(a, e, a') \in T_o$ if $a = a' \in A \land P_z(e, a) \neq 0$, or $a = a' = a_s \land P_z(e, a) \neq 0 \land \sum_{j \in A} P_z(e, j) \neq 1$, or $a = a_s \land a' \in A \land P_z(e, a) \neq 0$.
- $T_a = (A \times \{a_s\}) \cup (a_s, \phi)$ is the set of unobservable transitions from an annotated activity to $Idling$, and from $Idling$ to a deadlock location $\phi$.

**Semantics.** The state of the model is a triple $s = (\psi, \theta, \iota)$ where $\psi \in A_s$ is the current activity, $\theta \in \mathbb{R}_{\geq 0}$ is the remaining sojourn time, and $\iota \in \mathbb{R}_{\geq 0}$ is the remaining time to the next observable event. System evolution from a state $s = (\psi, \theta, \iota)$ depends on whether the current location is $a_s$ or an annotated activity, and $\theta$ is larger or smaller than $\iota$.

- **Case $\psi = a_s$.**
  **Event observed.** If $\theta > \iota$, an event $e \in E$ is observed with probability $P_z(e, a_s)$, and a new state $s' = (\psi', \theta', \iota')$ is selected randomly: with probability $P_z(e, a_s)$, an observable transition $(a_s, e, a)$ from $a_s$ to $a \in A$ occurs, in which case $\psi' = a_s$, while $\theta'$ and $\iota'$ are sampled from $F_{d_a}$ and $F_{i_a}$, respectively; with probability $1 - \sum_{a \in A} P_z(e, a)$, an unobservable transition $(a_s, e, a_s)$ occurs, so that $\psi' = a_s$, $\theta'$ is derived reducing $\theta$ by the time $\iota$ elapsed in $s$, and $\iota'$ is sampled from $F_{i_{a_s}}$.
  **Duration expired.** If $\theta < \iota$, the unobservable transition $(a_s, \phi)$ occurs and the deadlock state $s' = (\phi, \cdot, \cdot)$ is reached, from which no other transition is possible.

- **Case $\psi = a_s$.**
  **Event observed.** If $\theta > \iota$, an observable transition $(a, e, a)$ occurs from $a$ to itself with probability $P_z(e, a)$, yielding $s' = (a, \theta', \iota')$ where $\theta'$ is $\theta$ minus the time $\iota$ elapsed in $s$, and $\iota'$ is sampled from $F_{i_a}$.
  **Duration expired.** If $\theta < \iota$, an unobservable transition $(a, a_s)$ occurs, yielding $s' = (a_s, \theta', \iota')$ where $\theta'$ and $\iota'$ are sampled from $F_{d_{a_s}}$ and $F_{i_{a_s}}$, respectively.

At the initial time, the system is in state $s_0 = (\psi_0, \theta_0, \iota_0)$ where $\psi_0 = a_s$, and $\theta_0$ and $\iota_0$ are sampled from $F_{d_{a_s}}$ and $F_{i_{a_s}}$, respectively. Any annotated activity might be
selected as the initial location as well, without impacting on the complexity of the subsequent treatment.

**Running example.** The model in Fig. 2 has 2 annotated activities \(a_1\) and \(a_2\), and 4 observable events \(e_1, \ldots, e_4\), with activities depicted as circles, and transitions as solid or dashed arrows if they are observable or unobservable, respectively, with tick or thin line if they restart both the remaining inter-event time \(\ell\) and the remaining sojourn time \(\theta\), or \(\ell\) only, respectively. With \(\forall a \in A, \forall e \in E\), transitions \((a, e, a), (a_+, e, a), (a_+, e, a_+), (a, e, a)\) are depicted as solid thin arrows labeled with \((e, P_e(e), a)\), solid thin arrows labeled with \((e, P_e(e), 1 - \sum_{a \in A} P_e(e, a))\), solid tick arrows labeled with \((e, P_e(e), P_s(e)), (e, P_e(e), a)\), and dashed tick arrows, respectively. In Idling \(a_+\), events \(e_1, e_2, e_3\) occur with probability 0.35, 0.15, and 0.5, respectively; at \(e_1\) or \(e_2\) Idling \(a_+\) continues with probability 1; at \(e_3\) Idling \(a_+\) continues with probability 0.25, or \(a_1\) or \(a_2\) are started with probability 0.30 and 0.45, respectively. During \(a_1\), events \(e_1, e_2, e_4\) occur with probability 0.30, 0.60, and 0.10, respectively; besides, \(e_2, e_3, e_4\) occur during \(a_2\) with probability 0.35, 0.15, and 0.50, respectively.

Idling \(a_+\) has hyper-exponential sojourn time with PDF

\[
\begin{align*}
\quad f(x) &= \sum_{i=1}^{2} b_i \lambda_i e^{-\lambda_i x} \\
&= \sum_{i=1}^{2} p_{i,0} e^{-\lambda_i x} + \sum_{i=2}^{2} p_{i,2} e^{-\lambda_i x - \lambda_{i,2} x} + \sum_{i=2}^{2} p_{i,0} e^{-\lambda_i x}
\end{align*}
\]

(1) where \(\lambda_1 = 0.01, \lambda_2 = 0.02, p_{1,0} = 0.2, p_{2,0} = 0.8, p_{2,2} = 0.8, \) and exponential inter-event time with PDF \(f(x) = \lambda e^{-\lambda x}\) where \(\lambda = 0.05; \)

\(a_1\) has hyper-exponential sojourn time with \(\lambda_1 = 0.001, \lambda_2 = 0.005, p_{1,0} = 0.3, p_{2,0} = 0.2, \) and exponential inter-event time with \(\lambda = 0.03\); and \(a_2\) has hyper-exponential sojourn time with PDF \(f(x) = \lambda_1 x^2 (e^{-\lambda_1 x} - e^{-\lambda_2 x})\) where \(\lambda_1 = 0.05, \lambda_2 = 0.1, \) and exponential inter-event time with \(\lambda = 0.2\).

**Underlying stochastic process.** In any state, the remaining time to the end of the current activity \(\theta\) and to the next observable event \(\ell\) are concurrent and (possibly) generally distributed random variables, and thus the stochastic process \(X := \{s(t), t \in \mathbb{R}_{\geq 0}\}\), with \(s(t)\) denoting the state of the model at time \(t\), is not a Markov chain. At the observation of any event, \(\ell\) is restarted but \(\theta\) continues; but, always, with probability 1, an activity will eventually terminate and both \(\theta\) and \(\ell\) will be resampled; at that point, the future behavior depends on the past only through the current state (Markov condition), i.e., the process \(X \) regenerates. According to this, \(X\) is a Markov Regenerative Process (MRP) [33], [34]. Note that any number of events may be observed during an activity, thus the number of transitions between consecutive regenerations of \(X\) is unbounded. However, the number of transitions between observable events is not larger than 2: in any tagged activity, an unobservable transition can reach only Idling; in turn, in Idling, an unobservable transition can lead only to a deadlock.

While \(X\) is an MRP, the way it is used in combination with observations permits to manipulate it as an instance of the more restricted class of (continuous-time) Semi-Markov Processes (SMP) [33], [34]. In fact, at each observed event, the remaining time \(\theta\) to the end of the current activity continues, but the event timestamp associates the time elapsed since the previous state with a determined value \(\tau\). It is thus possible to associate \(\theta\) with a new distribution that makes the future behavior of \(X\) be dependent on the past only through the current state (Markov condition): if \(\theta_p\) is the random variable at the previous event and \(f_{\theta_p}\) is its PDF, \(\theta = (\theta_p - \tau)\theta_p > \tau\) and its PDF is thus \(f_\theta(x) = f_{\theta_p}(x + \tau) / \int_{\tau}^{\infty} f_{\theta_p}(y) dy\). For this reason, following the notation of [14], the model is termed Continuous Time Hidden Semi-Markov Model (CT-HSMM).

**B. Model enhancement**

The model is derived in closed form from the statistical abstraction of Section II-B, which can also be regarded as a case of model enhancement in process mining [26]. Specifically, \(A\) and \(E\) are equal to the sets \(A\) and \(E\), respectively, while \(P_s, P_x, P_y\), and \(P_e\) are equal to the statistics \(P_s, P_x, P_y, P_e\), respectively. Then, \(T_0\) is derived from \(A, E, P_s, P_x, P_y\), and \(P_e\) as the set that contains: an observable transition \((a, e, a) \forall a \in A\) and \(\forall e \in E\) such that \(P_e(e, a) \neq 0\); an observable transition \((a_+, e, a) \forall e \in E\) such that \(P_e(e, a) \neq 0 \land \sum_{a \in A} P_e(e, a)\); and, an observable transition \((a_+, e, a) \forall a \in A\) and \(\forall e \in E\) such that \(P_e(e, a) \neq 0\). Similarly, \(T_0\) is derived from \(A\) as the set that contains an unobservable transition \((a, a_+) \forall a \in A\).

Finally, the PDFs \(F_0(a)\) and \(F_1(a)\) of duration and inter-event time of each activity \(a \in A_0\) are defined so as to fit expected value \(\mu\) and coefficient of variation \(CV\) of corresponding histograms \(F_0(a)\) and \(F_1(a)\), respectively. To this end, we extend [31]: for histograms with \(CV < \frac{1}{\sqrt{2}}\), we replace the Shifted Exponential random variable (r.v.), which would produce a false negative for any value under a firm threshold, with the sum of an Exponential r.v. and an Erlang r.v. with sufficient phases to attain CV lower than samples. Specifically, if \(CV < 1\), the approximating PDF \(f(x)\) with \(x \in [0, \infty)\) is the PDF of the sum of an Erlang r.v. with shape \(n\) and rate \(\lambda_1\), and an Exponential r.v. with rate \(\lambda_2\), where \(n = [1/CV^2] - 1, \lambda_1\) is solution of \(CV^2 \mu^2 = \frac{n}{\lambda_1} + (\mu - n/\lambda_1)^2,\) and \(\lambda_2 = 1/(\mu - n/\lambda_1)\) (see Section II of Supplemental Material). If \(CV \approx 1, f(x) = \lambda e^{-\lambda x}\) with \(x \in [0, \infty)\) is the PDF of an Exponential r.v. with rate \(\lambda = \frac{1}{p}\) [31]. If \(CV > 1, f(x) = \sum_{i=1}^{n} p_i \lambda_i e^{-\lambda_i x}\) with \(x \in [0, \infty)\) is the PDF of a hyper-exponential r.v. with \(\lambda_i = \frac{2n}{\mu}\) and \(p_i = \frac{1}{\lambda}(\frac{CV^2 - 1}{CV^2 + 1})\) [31].

Note that, in the special case \(\frac{1}{\sqrt{2}} < CV < 1\), the proposed approximating PDF takes the same form as that of [31], and that the sequential composition of an Exponential r.v. and an Erlang r.v. conceptually follows the approach proposed in [35] to fit higher order moments. Finally, note that this class of

---

**Fig. 2.** A model with 2 annotated activities and 4 observable events. Sojourn and inter-event times, and the unobservable transition \((a_+, \phi)\), are not shown.
functions is used without loss of generality, as all the steps of the method and its implementation can be applied for any general distribution in the class of expolynomial functions.

IV. CLASSIFICATION TECHNIQUE

We define a diagnosis technique for online AR (Section IV-B) and offline AR (Section IV-C). To this end, transient analysis of the model of Section III is performed to derive the probability that an activity is performed at a given time conditioned on a sequence of time-stamped observations. The analysis can be efficiently performed through a tailoring of the method of stochastic state classes [29] (Section IV-A).

A. Preliminaries on stochastic state classes

A stochastic state class (class for short) represents the continuous set of states that can be reached after a sequence of (observable or unobservable) transitions. Specifically, a class is a tuple \( \Sigma = (\psi, D(\theta, t, \tau_{age})) \), where: \( \psi \in A \) is an activity; \( D(\theta, t, \tau_{age}) \subseteq \mathbb{R}_{\geq 0} \times \mathbb{R}_{\leq 0} \) is the support of the random vector \( (\theta, t, \tau_{age}) \), made of the remaining sojourn time \( \theta \) in \( \psi \), the remaining time \( t \) to the next observable event in \( \psi \), and the absolute time \( \tau_{age} \); and, \( f(\theta, t, \tau_{age}) : D(\theta, t, \tau_{age}) \rightarrow (0, \infty) \) is the PDF of \( (\theta, t, \tau_{age}) \) (as discussed in the Supplemental Material, \( \tau_{age} \) is the opposite of the elapsed time).

A succession relation \( \Sigma \xrightarrow{\mu} \Sigma' \) is defined between classes, meaning that \( \Sigma' \) is the successor of \( \Sigma \) through transition \( t \) with probability \( \mu \). Enumeration of the succession relation from an initial class \( \Sigma_0 \) yields a transient tree, where nodes are classes and edges are labeled with transitions and their occurrence probability. The reaching probability \( \eta(\Sigma_k) \) of a class \( \Sigma_k \) reached from \( \Sigma_0 \) through \( \Sigma_0 \xrightarrow{t_1\mu_1} \Sigma_1 \ldots \xrightarrow{t_k\mu_k} \Sigma_k \) is the product of transition probabilities, i.e., \( \eta(\Sigma_k) := \prod_{i=1}^{k} \mu_i \). Technical details on the calculus of successor classes are deferred to the Supplemental Material.

We consider an initial class \( \Sigma_0 = (a_0, \psi_0, D(\theta, t, \tau_{age})). \) \( \psi_0 \) where the logical location is the Idling activity \( a_0 \) and the random variables \( \theta, t, \) and \( \tau_{age} \) are independently distributed. Specifically, \( \theta \) and \( t \) are distributed according to \( F_\theta(a_0) \) and \( F_\psi(a_0) \) respectively, while \( \tau_{age} \) (measuring the elapsed time) is associated with the Dirac delta function centered at 0, i.e., \( D(\theta, t, \tau_{age}) := [\min F_\theta(a_0), \max F_\theta(a_0)] \times [\min F_\psi(a_0), \max F_\psi(a_0)] \times [0, 0] \) and \( f(\theta, t, \tau_{age}) := \delta(\tau_{age}) \).

B. Online classification

At any time \( t \), each activity \( a \in A \) has a measure of likelihood \( L^\text{on}(a, t, \Sigma_0, \omega_{1:n}) \) defined as the probability that \( a \) is performed at \( t \) conditioned on the initial class \( \Sigma_0 \) and on a sequence of observations \( \omega_{1:n} = (\omega_1, \ldots, \omega_n) \) taken within \([0, t]\), with \( \omega_i = (e_i, t_i) \forall i \in [1, n] \) and \( t_1 \leq t_2 \leq \ldots \leq t_n \leq t \).\footnote{To simplify notation, the type of the \( t \)-th event is indicated as \( e_i \) instead of \( e_{x(i)} \) as in Section II, avoiding the use of an indexing function \( x \).
}

\[ L^\text{on}(a, t, \Sigma_0, \omega_{1:n}) := P\{a(t) = a | \Sigma_0, \omega_1, \ldots, \omega_n\} \] (1)

The activity with the largest likelihood is the predicted class \( \rho^\text{on}(t) \), compared with the actual class \( \xi(t) \) tagged in the ground truth, i.e., \( \rho^\text{on}(t) := \arg\max_{a \in A} L^\text{on}(a, t, \Sigma_0, \omega_{1:n}) \).

At time \( t_0 \), no event of \( \omega_{1:n} \) has been observed yet, and the initial class \( \Sigma_0 \) collects the plausible states (see Section IV-A). According to this, \( \text{Idling} \) has likelihood 1, while any tagged activity has likelihood 0, i.e., \( L^\text{on}(a, t_0, \Sigma_0, \omega_{1:n}) = 1 \) and \( L^\text{on}(a, t_0, \Sigma_0, \omega_{1:n}) = 0 \forall a \in A \). At any time \( t > t_0 \), the likelihood of an activity is derived in a different manner depending on whether \( t \) is the time of an observation (Section IV-B1) or a time between consecutive observations (Section IV-B2).

1) Classification right after an observation: The likelihood of \( a \in A \) right after the \( n \)-th observation \( \omega_n = (e_n, t_n) \) is evaluated as the sum of the likelihoods of the classes with logical location \( a \) that can be reached from the initial class \( \Sigma_0 \) conditioned on the observation sequence \( \omega_{1:n} \):

\[ L^\text{on}(t_n, a, \Sigma_0, \omega_{1:n}) = \sum_{\Sigma \in \Gamma_n | \Sigma = \{a, \cdots\} \} \omega^\text{on}(\Sigma_{m}) \] (2)

where \( \omega^\text{on}(\Sigma_m) := P\{\Sigma_m | \Sigma_0, \omega_{1:n}\} \) is the probability that \( \Sigma_m \) is entered at time \( t_n \), conditioned on \( \Sigma_0 \) and \( \omega_{1:n} \), and \( \Gamma_n \) is the set of classes reached from \( \Sigma_0 \) through \( \omega_{1:n} \). According to the definition of conditional probability and to the law of total probability, \( \omega^\text{on}(\Sigma_m) \) can be expressed as:

\[ \omega^\text{on}(\Sigma_m) = \frac{\chi(\Sigma_0, \omega_{1:n}, \Sigma_m)}{\sum_{\Sigma_n \in \Gamma_n} \chi(\Sigma_0, \omega_{1:n}, \Sigma_n)} \] (3)

where \( \chi(\Sigma_0, \omega_{1:n}, \Sigma_m) := P\{\Sigma_m | \Sigma_0, \omega_{1:n}\} \) is a forward variable representing the probability that \( \Sigma_m \) is reached at \( t_n \) through \( \omega_{1:n} \) starting from \( \Sigma_0 \). If \( \Gamma_0 = \emptyset \), classification is restarted from \( \Sigma_0 \), i.e., \( \Gamma_n = \{\Sigma_0\} \) and \( \omega^\text{on}(\Sigma_0) = 1 \).

In the following, we first derive the set of classes \( \Gamma_n \) and then we evaluate \( \chi(\Sigma_0, \omega_{1:n}, \Sigma_m) \) for each class \( \Sigma_m \in \Gamma_n \).

1.1) Derivation of set \( \Gamma_n \) of classes after \( n \) observations. \( \Gamma_n \) consists of \( \Sigma_0 \), i.e., \( \Gamma_0 = \{\Sigma_0\} \). When \( \omega_1 = (e_1, t_1) \) occurs, \( \Sigma_0 \) is conditioned on the occurrence of \( e_1 \) at \( t_1 \), yielding \( \Gamma_1 = \{\Sigma_1, \Sigma_2, \ldots, \Sigma_{M_1}\} \), e.g., in Fig. 3, \( \Gamma_1 = \{\Sigma_1, \Sigma_2, \Sigma_3\} \). In so doing, when \( \omega_n = (e_n, t_n) \) is observed, each class in \( \Gamma_{n-1} = \{\Sigma_{n-1}, \Sigma_{n-2}, \ldots, \Sigma_{M_{n-1}}\} \) is conditioned on the occurrence of \( e_n \) at \( t_n \), yielding \( \Gamma_n = \{\Sigma_n, \Sigma_2, \ldots, \Sigma_{M_n}\} \).

As shown in Fig. 3, it may be the case that: i) a class in \( \Gamma_{n-1} \) yields two or more classes in \( \Gamma_n \), which may occur, for instance, when the class in \( \Gamma_{n-1} \) has the Idling location and the occurrence of \( e_n \) may represent the continuation of Idling or start of multiple annotated activities, (e.g., \( \Sigma_0 \) yields \( \Sigma_1 \), \( \Sigma_2 \), and \( \Sigma_3 \)); ii) two or more classes in \( \Gamma_{n-1} \) yield the same class in \( \Gamma_n \), which occurs, for instance, when those classes in \( \Gamma_{n-1} \) have different activity and the occurrence of \( e_n \) represents the continuation of Idling after the completion of the activity, (e.g., \( \Sigma_2 \) and \( \Sigma_3 \) yield \( \Sigma_4 \)); iii) a class in \( \Gamma_{n-1} \) yields no class in \( \Sigma_n \), which occurs when \( e_n \) at \( t_n \) is not possible in any activity allowed by that class of \( \Gamma_{n-1} \) (e.g., class \( \Sigma_4 \)).

Hence, conditioning each class \( \Sigma_{m_{n-1}} \in \Gamma_{n-1} \), with \( m \in [1, M_{n-1}] \), on the occurrence of \( e_n \) at time \( t_n \), yields a (possibly empty) set of classes \( \Gamma_m = e_n, t_n \), so that \( \Gamma_n = \bigcup_{m=1}^{M_{n-1}} \Gamma_m, e_n, t_n \).

The conditioning is developed in two steps:

- Conditioning on the observed event \( e_n \). Each class \( \Sigma_{m_{n-1}} \in \Gamma_{n-1} \), with \( m \in [1, M_{n-1}] \), is conditioned
on the observation of \( e_n \), yielding the set \( \Gamma^{m,e_n} \) of classes: i) are leaves of the transient tree enumerated from \( \Sigma^m_{n-1} \) up to the next observable events (i.e., enumeration is halted if an observable transition or the unobservable transition \( (\theta, \iota) \) occurs), and ii) are reached by an observable transition \( (\cdot, e_n, \cdot) \). Note that the tree is finite, given that at most two unobservable transitions may occur before any observable event, as remarked in Section III-A. In Fig. 4, the observation of \( \omega_2 = (e_4, 400 \text{s}) \) makes plausible only the hypothesis that \( a_2 \) is continued, leading to \( \Gamma^{e_2,\omega} = \{ \Sigma_{2d} \} \).

• Conditioning on the observed time \( t_1 \). In turn, each class \( \Sigma_{m,e_n,k} \) in \( \Gamma^{1,m,e_n} \), with \( k \in [1, \Gamma^{m,e_n} \cdot 1 \text{,}\, \lambda] \), is conditioned on the observation time \( t_1 \) by imposing that the time elapsed between the observations \( \omega_{n-1} \) and \( \omega_1 \) is equal to \( t_n - t_{n-1} \) (i.e., \( -\tau_\text{age} = t_n - t_{n-1} \)), yielding a class \( \Sigma_{m,e_n,k} \). In each class \( \Sigma_{m,e_n,k} \), \( \theta \) and \( \iota \) are independently distributed, so that the support and the PDF of \( \tau_\text{age} \) can be reset to \([0,0]\) and \( \delta(x_\text{age}) \), respectively, as proved by the following Theorem. The set of classes \( \Gamma^{m,e_n,t_1} \) is finally derived as \( \Gamma^{m,e_n,t_1} = \bigcup_{k=1}^{\Gamma^{m,e_n} \cdot 1 \text{,}\, \lambda} (\Sigma_{m,e_n,k} \cdot \iota) \).

For instance, conditioning class \( \Sigma_{3} \) in Fig. 4 on the time of observation \( \omega_2 = (e_4, 400 \text{s}) \) yields class \( \Sigma_{2d} \) in Fig. 3.

**Theorem 1.** \( \forall \) class \( \Sigma = (\psi, D(\theta_\text{age}, \iota), f(\theta_\text{age}, \iota)) \in \Gamma_n \), with \( n > 0 \), \( \forall \) random variables \( \theta_\text{age} \) and \( \iota_\text{age} \) are independently distributed, and the random vector \( (\theta_\text{age}, \iota_\text{age}) \) has domain \( D_{(\theta_\text{age}, \iota_\text{age})} = D_\theta \times [\Delta_{\min}, \Delta_{\max}] \times [0,0] \) and PDF \( f_{(\theta_\text{age}, \iota_\text{age})}(x_\theta, x_\iota, \tau_\text{age}) = f_\theta(x_\theta) f_\iota(x_\iota) \delta(x_\text{age}) \), where \( D_\theta \) and \( f_\theta \) are the marginal domain and the marginal PDF of \( \theta \), respectively. If entering class \( \Sigma \) accounts for starting activity \( \psi_1 \), then \( D_\psi = [\Delta_{\min}, \Delta_{\max}] \) and \( f_\psi(x) = f_\psi(x) \). Otherwise (i.e., if entering \( \Sigma \) represents a continuation of \( \psi_1 \)), \( D_\psi = \{ x_\theta, x_\iota, x_\Delta \in D_\psi, x_\Delta \geq \Delta \} \) and \( f_\psi(x) = f_\psi(x_\theta + \Delta_\text{age}) = \int_{\Delta_\text{age}} f_{\psi}(y) \text{d}y \), where \( \Delta_\text{age} = t_n - t_{n-1} \), and \( D_\psi \) and \( f_\psi \) are the domain and the marginal PDF of \( \Delta_\text{age} \), respectively, of the remaining sojourn time \( \theta_\text{age} \) in the class \( \Sigma_\psi \in \Gamma_n \) (at the previous event) that yields \( \Sigma \).

For instance, in Fig. 3, the PDF of \( \theta \) in \( \Sigma_1 \) is \( f_\theta(a_1)(x_\theta) \), while in \( \Sigma_2 \) it is \( f_\theta(a_1)(x_\theta + 220) + \int_{220} f_\theta(a_1)(y) \text{d}y \). Theorem 1 is instrumental to prove that \( \Gamma_n \) is finite \( \forall n > 0 \).

---

**Theorem 2.** \( \forall n > 0 \), the number of classes in \( \Gamma_n \) is bounded by \( Q_{max} = (|A|+1) \cdot 2^C \), where \(|A|\) is the number of annotated activities, and \( C \) is the maximum number of consecutive events that may represent the start or the continuation of an activity.

1.2) Derivation of the forward variable \( \chi(\Sigma_0, \omega_{1,n}, \Sigma_n^m) \).

**Theorem 3.** \( \forall n > 0 \) and \( \forall \Sigma_n^m \in \Gamma_n > 0 \), it holds that:

\[
\chi(\Sigma_0, \omega_{1,n}, \Sigma_n^m) = \sum_{\Sigma_{n-1} \in \Gamma_{n,\text{parent}}} \chi(\Sigma_0, \omega_{1,n-1}, \Sigma_n^{i}) \cdot P(\omega_n, \Sigma_n^m | \Sigma_{n-1})
\]

where: \( \Gamma_{n,\text{parent}} \subseteq \Gamma_n \) is the set of classes that, conditioned on \( \omega_{n-1} = (e_1, t_{n-1}, \iota_1) \), yield \( \Sigma_{n-1} \). \( P(\omega_n, \Sigma_n^m | \Sigma_{n-1}) \) is the probability that \( \omega_n \) is observed and class \( \Sigma_n^m \) is reached at time \( t_n \) given that class \( \Sigma_{n-1} \) was entered at time \( t_{n-1} \); and \( \chi(\Sigma_0, \omega_{1,0}, \iota_0) = 1 \) is the probability of the initial class.

In turn, \( P(\omega_n, \Sigma_n | \Sigma_{n-1}) \) can be derived as:

\[
P(\omega_n, \Sigma_n | \Sigma_{n-1}) = \int_{t_n} f^{\text{leaf}}(x_\text{age}) \text{d}x_\text{age}.
\]

The first factor is the reaching probability of the class \( \Sigma_n^{\text{leaf}} \) that: i) is a leaf in the tree enumerated from \( \Sigma_{n-1} \in \Gamma_{n,\text{parent}} \) up to any observable event, ii) is reached by event \( e_n \), and iii) yields \( \Sigma_n \) if conditioned on the observation event \( t_n \). The second factor is the integral of the marginal PDF of \( x_\text{age} \) in \( \Sigma_n^{\text{leaf}} \) over \( t_n = [\Delta_n - \frac{\pi}{2}, \Delta_n + \frac{\pi}{2}] \), where \( \Delta_n = t_n - t_{n-1} \) and \( \delta \) is the minimum time between consecutive events. Hence, the time of observations is taken into account, e.g., though in the model of Fig. 2 the probability to observe \( e_4 \) is larger during \( a_2 \) than during \( a_1 \), in Fig. 3 the online likelihood of \( \Sigma_2 \) is lower than that of \( \Sigma_1 \) due to the fact that the probability to observe \( e_4 \) in a neighborhood of time \( 220 \text{s} \) (since the activity start) is significantly larger in \( a_1 \) than in \( a_2 \).

2) Classification between consecutive observations: In the interval \( [t_n, t_{n+1}] \) between observations \( \omega_n \) and \( \omega_{n+1} \), transitive analysis [29] is performed from each class \( \Sigma_n^m \in \Gamma_n \) up to the next observable events with time limit \( \Delta_n \), yielding \( \forall a \in A \), the probability \( p(a, t, \Sigma_n^m) := P(a | \Sigma_n^m) \) that \( a \) is performed at time \( t \in (t_n, t_{n+1}) \) conditioned on the

---

![Fig. 3](image-url)  
**Fig. 3.** The sets \( \Gamma_1 = \{ \Sigma_1^1, \Sigma_1^2, \Sigma_1^3 \} \), \( \Gamma_2 = \{ \Sigma_2^1, \Sigma_2^2, \Sigma_2^3 \} \), \( \Gamma_3 = \{ \Sigma_3^1, \Sigma_3^2, \Sigma_3^3 \} \) of plausible classes for the model of Fig. 2 after observations \( \omega_2 = (e_2, 180 \text{s}) \), \( \omega_2 = (e_4, 400 \text{s}) \), \( \omega_3 = (e_1, 420 \text{s}) \), respectively. Each class \( \Sigma \) is labeled with its online likelihood \( \text{ln} \bigl( p(a, t, \Sigma_n^m) \bigr) \) (in plain text) and offline likelihood \( \text{off} \bigl( p(a, t, \Sigma_n^m) \bigr) \) (in bold), and filled with gray or white if the corresponding activity is started or continued when the class is entered, respectively.

![Fig. 4](image-url)  
**Fig. 4.** Transient tree of the model of Fig. 2 enumerated from class \( \Sigma_2^2 \) of Fig. 3 up to the next observable events (only observable transitions are shown, using the same graphical notation of Fig. 2). Class \( \Sigma_2^{2d} \) reached through the observation of \( \omega_2 = (e_4, 400 \text{s}) \) is filled with light gray.
fact that $\Sigma^m_n$ was entered at time $t_n$. Then, the likelihood of activity $a$ is derived from $p(a, t, \Sigma^m_n)$ through Theorem 4.

**Theorem 4.** $\forall n > 0$ and $\forall t \in (t_n, t_{n+1})$, it holds that:

$$L^n(a, t, \Sigma_0, \omega^1_{1,n}) = \sum_{\Sigma^m_n \in \Gamma_n} p(a, t, \Sigma^m_n) L^n(\Sigma^m_n).$$

(6)

C. Offline classification

Given a sequence of observations $\omega^1_{1,N}=\langle \omega_1, \ldots, \omega_N \rangle$, with $\omega_i = (e_i, t_i) \forall i \in [1,N]$ and $t_1 \leq t_2 \leq \ldots \leq t_N$, at any time $t \in [0, t_N]$ each activity $a \in A_*$ is associated with a measure of likelihood $L^n(\omega^1_{1,N})$ defined as the probability that $a$ is performed at time $t$ on the overall observation sequence $\omega^1_{1,N}$, considering not only the observations taken within the interval $[0, t]$ (as in the online classification) but also those taken within the interval $(t, t_N]$:

$$L^n(t, a, \Sigma_0, \omega^1_{1,N}) := P\{a(t) = a | \omega_1, \ldots, \omega_N\} \tag{7}$$

As for the online classification, at any time $t$, the predicted class $\rho^{\text{off}}(t)$ is the activity with the largest measure of likelihood, i.e.,

$$\rho^{\text{off}}(t) := \arg\max_{a \in A^*} L^n(a, t, \Sigma_0, \omega^1_{1,N}),$$

derived in a different manner depending on whether $t$ is the time instant of an observation (Section IV-C1) or a time instant between two consecutive observations (Section IV-C2).

1) Classification right after an observation: The likelihood of $a \in A_*$ right after the $n$-th observation $\omega_n = (e_n, t_n)$ is evaluated as the sum of the likelihoods of the classes with the same location $a$ that can be reached from the initial class $\Sigma_0$ conditioned on the entire observation sequence $\omega^1_{1:N}$:

$$L^n(t_n, a, \Sigma_0, \omega^1_{1,N}) = \sum_{\Sigma \in \Gamma_n} l^{\text{off}}(\Sigma^m_n) \tag{8}$$

where $l^{\text{off}}(\Sigma^m_n) := P\{\Sigma^m_n | \Sigma_0, \omega^1_{1,N}\}$ is defined as the probability that $\Sigma^m_n$ is entered at time $t_n$ conditioned on $\Sigma_0$ and $\omega^1_{1,N}$, and can be evaluated through a kind of Forward-Backward procedure [4] according to Theorem 5.

**Theorem 5.** $\forall n > 0$ and $\forall \Sigma^m_n \in \Gamma_n$, it holds that:

$$l^{\text{off}}(\Sigma^m_n) = \frac{\chi(\Sigma_0, \omega^1_{1:n}, \Sigma^m_n) \beta(\Sigma^m_n, \omega^1_{1:n+1,N})}{\sum_{\Sigma \in \Gamma_n} \chi(\Sigma_0, \omega^1_{1:n}, \Sigma) \beta(\Sigma, \omega^1_{1:n+1,N})}, \tag{9}$$

where $\beta(\Sigma^m_n, \omega^1_{1:n+1,N}) := P\{\omega^1_{1:n+1,N} | \Sigma^m_n\}$ is a backward variable defined as the probability of the observation sequence $\omega^1_{1:n+1,N}$ conditioned on the fact that $\Sigma^m_n$ is entered at time $t_n$.

In turn, $\beta(\Sigma^m_n, \omega^1_{1:n+1,N})$ is derived iteratively by Theorem 6.

**Theorem 6.** $\forall 0 < n < N$ and $\forall \Sigma^m_n \in \Gamma_n > 0$, it holds that:

$$\beta(\Sigma^m_n, \omega^1_{1:n+1,N}) = \sum_{\Sigma^m_0 \in \Gamma^\text{child}_{n,m}} \beta(\Sigma^m_0, \omega^1_{1:n+2,N}) \cdot P\{\omega^1_{1:n+1}, \Sigma^m_0 | \Sigma^m_n\} \tag{10}$$

where: $\Gamma^\text{child}_{n,m} \subseteq \Gamma_{n+1}$ is the set of classes yielded by $\Sigma^m_n$ conditioned on observation $\omega^1_{n+1}$, $P(\omega^1_{n+1}, \Sigma^m_{n+1} | \Sigma^m_n)$ is derived through Eq. (5); and, $\beta(\Sigma_N, \omega^1_{1:N+1}) = 1 \forall \Sigma_N \in \Gamma_N$.

D. Complexity

The complexity of CT-HSMMs linearly depends on the number $L_{\text{ct}}$ of observed events rather than on the number $L$ of time ticks at which classification is emitted, as in HSMMs. Note that $L$ may be huge to attain sufficient granularity when temporal parameters of the model have different time scale, and typically it is significantly larger than $L_{\text{ct}}$, e.g., in Section V, we consider time ticks of 60 s as in [36] to compare with HSMMs, yielding $L = 40,320$, against $L_{\text{ct}} = 1319$.

After each event, the set $\Gamma'$ of plausible classes is derived from the set $\Gamma$ of classes that were plausible after the previous event. For $U \leq |A|$ classes in $\Gamma'$ representing the start of an annotated activity, the successor classes in $\Gamma'$ and the corresponding transition probabilities can be pre-computed. For the remaining $V \geq |A| - |A|$ classes, the logical location of each successor class in $\Gamma'$ and the PDF of the remaining sojourn time $\theta$ in successor classes representing the start of an annotated activity can be derived a-priori, while all transition probabilities and the PDF of $\theta$ in successor classes representing the continuation of an activity must be computed based on the observed time. Given that each class in $\Gamma$ yields at most $|A| + 2$ classes in $\Gamma'$, the complexity is proportional to $|V - U|(|A| + 2) \approx Q_{\text{max}}|A|$, where $Q_{\text{max}}$ is the maximum number of plausible classes.

Moreover, after each event, transient analysis is performed from each plausible class until the next event. Probabilities are computed in a constant number of time points, yielding linear complexity in $Q_{\text{max}}$. Note that the number of time points could be significantly reduced, e.g., not performing the analysis if the most likely activity after the next event remains the same.

By Theorem 2, $Q_{\text{max}} = (|A| + 1)^2 C$, where $C$ is the maximum number of consecutive events being the start or continuation of an activity, yielding overall complexity $O(|A|^2 (L_{ct} + 1))$. As typical in continuous-time solution techniques, worst case analysis of complexity provides overly conservative results: Theorem 2 guarantees termination, but in principle indicates an exponential complexity. In practice, the number of concurrent plausible classes turns out to be much lower than $Q_{\text{max}}$, e.g., in the experiments of Section V, the number of plausible classes per day is at most 225 and 69 on average. Note that, to improve scalability, the actual cost could be largely reduced by pruning hypotheses with low likelihood, e.g., discarding classes with likelihood at least one order of magnitude lower than that of other classes with the same logical location.

V. Computational Experience

A. Description of the data set under consideration

The approach is experimented on a publicly available data set [6] containing: 1319 raw events collected by 14 state-change sensors deployed at different locations in a 3-room
apartment (e.g., doors and household appliances); 245 activities of 7 distinct types (i.e., Preparing breakfast, Preparing a beverage, Preparing dinner, Taking shower, Toileting, Sleeping, and Leaving house) plus Idling, performed and manually annotated over a period of 28 days. Activities are annotated in a mutually exclusive way (i.e., one activity at a time), with the only exception of a few instances of Toileting nested in Sleeping (21 instances) or in Preparing dinner (1 instance).

B. Experimental setup

The data set is split into training and test sets according to a leave one day out approach, using sensor data of each day for testing and those of the remaining days for training (as discussed in Section II, events natively encoded by sensors using the raw representation are converted into a dual change-point representation). Events observed during the periods of absence of the subject (i.e., Leaving house) are removed to avoid inconsistencies (except for Opening/Closing front door).

A prototype implementation based on the API of the ORIS tool [37], [38] was used to perform experiments for both the proposed online and offline classification techniques, considering both non-Markovian durations described in Section III-B, and exponentially distributed durations that fit only the expected value. In the latter case, the model has an underlying stochastic process in the class of Continuous Time Markov Chains (CTMC) and becomes conceptually close to CT-HMMs of [14], with a notable difference: using CT-HMMs, each state would be made of an activity and an observed event. This increases the expressivity, permitting the next transition and the rate of the (exponential) sojourn time to be dependent also on the last observed event. However, this also increases the number of states, with impact on computational complexity and, more importantly, on the volume of data needed to avoid overfitting. Despite this difference, for notational convenience, in the rest of this Section, CT-HSMMs with exponential durations and inter-event times will be referred to as CT-HMM.

Results achieved by CT-HSMMs and CT-HMMs are compared with those attained by two discrete-time generative models, namely offline HMMs and offline HSMMs, running the implementation used in [36] on the considered data set. To permit a fair comparison, the predicted activity \( \rho(t) \) is compared with the ground truth activity \( \xi(t) \) using time slices of duration of 60 s as in [36]. For each activity \( a \in A_* \), we obtain the number of time slices representing True Positives (TP\(_a\)), False Positives (FP\(_a\)), and False Negatives (FN\(_a\)), permitting evaluation of precision \( Pr_a = TP_a/(TP_a + FP_a) \) (i.e., number of correctly predicted time slices for activity \( a \) out of the total number of time slices predicted for \( a \)), and recall \( Re_a = TP_a/(TP_a + FN_a) \) (i.e., number of correctly predicted time slices for \( a \) out of the total number of time slices of \( a \)).

Two aggregated performance measures are considered, i.e., the accuracy \( Acc = \sum_{a \in A_*} TP_a/\sum_{a \in A_*}(TP_a + FN_a) \) (i.e., the number of correctly predicted time slices out of the total number of time slices) and the average recall \( Re = \sum_{a \in A_*} Re_a/|A_*| \) (i.e., the average of all recall values).

Moreover, the following classification errors are considered for each activity: i) never recognized (NR), when all the time slices of the activity are misclassified; ii) latest started (LS), when the initial time slices of the activity are misclassified but all the subsequent ones are correctly classified; iii) early ended (EE), when the last time slices of the activity are misclassified but all the previous ones are correctly classified; and, iv) interrupted (IN), when misclassified and correctly classified time slices are interleaved. In the limited number of cases where an occurrence of Toileting is nested in an occurrence of Sleeping or Preparing dinner, the nested activity is removed from the ground truth, and thus a time-slice is correctly classified if the predicted activity type is equal to Sleeping or Preparing dinner, respectively.

C. Experimental results

1) Performance measures: Table I compares accuracy and average recall achieved by the proposed techniques and by offline HMMs and offline HSMMs. In particular, the reference method proposed in this paper is offline CT-HSMMs, a continuous-time approach which is based on a model with non-Markovian durations and implements offline classification: it achieves an accuracy of 0.956 and an average recall of 0.685.

Continuous-time vs discrete-time. On the one hand, offline CT-HSMMs can be compared with the corresponding discrete-time approach of offline HMMs, which achieve an accuracy of 0.761 and an average recall of 0.691. According to this, offline CT-HSMMs attain slightly lower average recall than offline HMMs, while significantly gaining in accuracy.

Non-Markovian vs Markovian. On the other hand, offline CT-HMMs can be compared with offline CT-HMMs, which achieve an accuracy of 0.956 and an average recall of 0.648, showing that the use of non-Markovian durations slightly improves average recall with respect to offline CT-HSMMs.

In the discrete-time domain, the comparison between the non-Markovian approach of offline HSMMs and the Markovian approach of offline HMMs yields a counterintuitive outcome: offline HMMs achieve an accuracy of 0.895 and an average recall of 0.739, which definitely outperform the 0.761 and the 0.691 attained by offline HSMMs, respectively. This result may be due to the limited volume of data available for training, which may cause overfitting, and to implementation choices requested by the discrete-time approach, e.g., in [37], time is discretized into ticks of duration equal to 60 s and distributions are represented as histograms with 5 bins.

It is remarkable that performance measures attained by offline HMMs can be compared also with those of offline CT-HMMs, showing that, as in the comparison of offline HMMs with offline CT-HSMMs, the continuous-time approach improves accuracy while showing lower average recall.

Offline vs online. Online CT-HSMMs achieve an accuracy of 0.953 and an average recall of 0.668, which compare with 0.956 and 0.685 attained by offline CT-HSMMs, respectively, while online CT-HMMs achieve an accuracy of 0.957 and an average recall of 0.626, which compare with 0.956 and 0.648 attained by offline CT-HMMs, respectively. Intuitively, exploiting only the forward variable reduces the available information and produces a (minor) reduction in performance measures, which however affects average recall but not accuracy.
Remark. Overall, the proposed continuous-time techniques improve accuracy of the considered discrete-time approaches, while showing lower average recall. Notably, the proposed methods provide consistent results: non-Markovian techniques improve on Markovian techniques (i.e., CT-HSMMs improve on CT-HMMs, both in the online and in the offline version), and offline techniques improve on online techniques (i.e., offline CT-HSMMs and offline CT-HMMs improve on online CT-HSMMs and online CT-HMMs, respectively). This result is not achieved by the discrete-time approaches, with offline HMMs definitely outperforming offline HSMMs.

2) Insight and discussion: Table II shows performance measures per class, highlighting particularly low precision values for Preparing a beverage and Preparing dinner. Confusion matrices, also reported in Table II, show that these wrong predictions mainly depend on low recall values of Idling, due to a non-negligible number of time slices misclassified as Preparing a beverage and Preparing dinner (e.g., 10 and 440 for offline CT-HSMMs, respectively). These misclassifications have no significant impact on the recall of Idling, which is the third most frequent class (i.e. 4974 time slices in the ground truth), but reduce the precision of Preparing a beverage and Preparing dinner due to their lower volume (i.e. only 17 and 342 time slices in the ground truth, respectively). Table II shows also particularly low recall values for Preparing a beverage and Preparing dinner due to misclassifications as Idling (e.g., 4 and 128 time slices of Preparing a beverage and Preparing dinner are misclassified as Idling by offline CT-HSMMs, respectively), and also due to a significant number of time slices of Preparing a beverage misclassified as Preparing dinner (e.g., 6 for offline CT-HSMMs). Overall, misclassifications of Preparing a beverage, Preparing dinner, and Idling may be due to the ambiguity associated with events that may start or continue multiple activities, e.g., Opening fridge may be the initial event of both Preparing a beverage and Preparing dinner, while Starting dishwasher may be a continuing event of both the activities. Moreover, these classification errors may be also due to the fact that the proposed model of CT-HMMs and CT-HSMMs distinguishes the starting event of an activity but not the concluding event.

Table III provides further insight showing the number of classification errors per activity. In more than half of the cases, misclassifications of Preparing a beverage, Preparing dinner, and Idling by online CT-HSMMs are due to NR or IN errors. Though, as expected, the number of IN errors is significantly reduced using offline CT-HSMMs, NR and IN errors comprise a structural limitation of the proposed approach.

Conversely, LS and EE errors inherently depend on the fact that the duration of an activity is evaluated as the time elapsed from its first event to its last event, as illustrated in Section II-B. In principle, these errors could be reduced by characterizing the random variables representing the time elapsed from the start of an activity to its first event and from the last event of an activity to its completion, respectively. Though this model variant is expected to reduce the number of time slices misclassified as Idling, it would make the computed statistics dependent on the ground truth, which would be used not only in a qualitative perspective to identify the performed activity, but also in a quantitative perspective to determine its duration. Given that ground truth annotations are structurally imprecise, and a completely supervised data set may not be always available, this solution would weaken the approach.

VI. CONCLUSIONS

We presented a continuous-time approach for online and offline AR in AAL, based on a model where durations are either Markovian (CT-HMMs) or non-Markovian (CT-HSMMs). Experiments performed on a real data set [6] prove feasibility of the approach, showing that offline CT-HMMs achieve better accuracy but lower average recall than offline HMMs, while offline CT-HSMMs attain largely better accuracy and comparable average recall with respect to offline HSMMs.

Achieving comparable performance with respect to state-of-the-art discrete-time machine learning techniques is valuable, since the proposed continuous-time approach is more general, being equivalently applicable when the observed phenomena...
are asynchronous or have different timescales. In fact, the approach avoids time discretization, which would require the achievement of a trade-off between granularity and complexity, and may become critical when temporal parameters have different orders of magnitude. Moreover, while parameter optimization in machine learning typically requires a large amount of data, model enhancement in the proposed continuous-time setting is less exposed to overfitting, since only few parameters have to be fit (e.g., the first two moments). Indeed also the proposed approach would benefit of the availability of more data, which would enable the derivation of finer approximants.

Insight gained on results opens the way to many extensions. Specifically, misclassifications due to activities characterized by the same events could be reduced extending the model with the notion of *time of day* by the same events could be reduced extending the model with data, which would enable the derivation of finer approximants.

The approach is open to extension to a fully data-driven model elicitation, which would permit application to cases where domain knowledge is not available or requires a long and expensive acquisition process. Conversely, extension to unsupervised data sets and concurrent activities would require major advancements both in the modeling and in the solution.

REFERENCES


Marco Biagi received the M.S. in Computer Engineering (2011) and the B.S. in Computer Engineering (2008) from the University of Florence, Italy. He has been a Ph.D. student in the Smart Computing program at University of Florence, Italy. His research activity mainly focused on quantitative approaches for diagnosis, prediction and action scheduling in partially observable systems. He is currently a senior software engineer in a video advertising company.

Laura Carnevali is Assistant Professor of Computer Science (with tenure track) at the School of Engineering, University of Florence, Italy. She received the Ph.D. in Informatics, Multimedia, and Telecommunications Engineering (2010) and the M.S. in Computer Engineering (2006) from the University of Florence, Italy. Her research is focused on correctness verification and quantitative evaluation of real-time and concurrent systems.

Marco Paolieri is a senior research associate at the University of Southern California, Los Angeles. He received his Ph.D. in Computer Science, Systems and Telecommunications (2015) and his M.S. in Computer Engineering (2011) from the University of Florence, Italy. His research interests focus on stochastic modeling and quantitative evaluation of performance and reliability in concurrent and distributed systems.

Fulvio Patara is a postdoctoral research fellow at the University of Florence, Italy, where he received his Ph.D. in Computer Science, Systems and Telecommunications (2016) and his M.S. in Computer Engineering (2011). Since 2018 he is Adjunct Professor of Software Engineering at the School of Engineering, University of Florence, Italy. His research is presently focused on stochastic modeling for Activity Recognition in Ambient Assisted Living and performance evaluation of multi-level metamodeling architectures applied to e-Health.

Enrico Vicario (M ’95) received the Ph.D. in Informatics and Telecommunications Engineering (1994) and the M.S. in Electronics Engineering (1990) from the University of Florence, Italy. He is Full Professor of Computer Science and Head of the Information Engineering Department at the School of Engineering, University of Florence, Italy. His research is presently focused on model based approaches for development, verification, and quantitative evaluation of concurrent systems with non-deterministic and stochastic temporal parameters.