Applying Activity-Based Models to Integrate Labeled Preset Key Events in Intra-Day Human Mobility Scenarios

Patrik Gonçalves[1,a] and Harald Baier[2,b]

[1]*Zentrale Stelle für Informationstechnik im Sicherheitsbereich, Zamdorfer Straße 88, München, Germany*

[2]*Research Institute Cyber Defence (CODE), Universität der Bundeswehr München, Carl-Wery-Straße 22, Munchen, Germany*

Keywords: Activity-Based Models, Non-Typical Behavior, Preset, Custom, A-Priori, Activities, Key Events, Outliers, Human Mobility, Dataset, Intra-Day.

Abstract: The generation of synthetic human mobility scenarios is often realized through data-driven or rule-based approaches. They work in a fire-and-forget principle and provide limited support to induce controlled activities in simulated scenarios. However, including controlled preset activities in the generation phase enables the creation of mobility scenarios that include a-priori known outliers or key events. Such mobility test datasets might be used in outlier detection for machine learning algorithms or for inducing non-typical mobility, where models do not exist or are too complex to construct. In this work we propose an activity-based scheduler to include controlled preset key events in the scheduling process of daily human mobility scenarios. Further, with our rule-based approach we can synthesize new activities of a target region even when initial data is unavailable or missing. In addition we propose a hierarchical methodology to iteratively add activities according to their number of constraints and provide a publicly available Python-based implementation. Our validation shows that our approach is able to integrate non-typical behavior in typical mobility scenarios.

1 INTRODUCTION

Contemporary mobile devices such as smartphones, smart wearables, GPS navigation systems or Internet of Things devices are able to determine a device's position in real time. By storing the device's current position over long time periods, human individuals wearing or using these devices can be tracked in a very convenient way. The resulting human-based geospatio-temporal movement is referred to as *human mobility* (Barbosa et al., 2018; Castiglione et al., 2015).

Geospatial datasets of human mobility are bound to specific regions and describe a specific pattern according to the used modes of transportation. Therefore, the transferability to other regions and modes is limited (Luca et al., 2021), hence a researcher needs to gather or synthesize new data for the target region fitting to his use case. Current data-driven approaches, e.g., machine learning (ML)-based solutions like (Bösche et al., 2012; Drchal et al., 2019), still need a representative dataset in the training phase, which on the other hand is often acquired from mobility surveys and custom-acquired tracking data of the target region. On the other hand, these datasets often suffer from weaknesses, which render their use in practice difficult to impossible. For instance, if a dataset is anonymized, proprietary (e.g., with respect to a closed format or high costs), unlabeled, unrealistic, unsuitable for the individual use case (i.e., the dataset represents an incorrect mobility pattern for the respect application) or in the worst case not complete (Aschenbruck et al., 2010).

Applying human mobility models is a convenient way to generate synthetic data on a large scale, where the other options fail due to spatial restrictions (e.g., creating tracking data of a large area or collecting data from a restricted area), resource limitations (e.g., limited in time or funding) or legal constraints (e.g., real human mobility data is considered personal data, which is restricted by privacy laws under the European Data Regulation (European Parliament and Council, 2016)).

The human mobility models describe typical behavior, e.g., according to a training dataset or to pre-

[a] https://orcid.org/0000-0001-5561-8818

[b] https://orcid.org/0000-0002-9254-6398

Gonçalves, P. and Baier, H.
Applying Activity-Based Models to Integrate Labeled Preset Key Events in Intra-Day Human Mobility Scenarios.
DOI: 10.5220/0011850000003479
In *Proceedings of the 9th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2023)*, pages 281-288
ISBN: 978-989-758-652-1; ISSN: 2184-495X

defined rules. On the other hand, when individuals decide to alter their normal behavior, it might not be considered in these models. For instance, from the psychological forensics perspective, criminal behavior is hard to model as criminals adapt to their environment by changing their actions to elude prosecution, as described by Holmes and Holmes (Holmes and Holmes, 2008). From another view, in machine learning algorithms the outliers describe disparate data that opposes the data from the trained models, i.e. anomalies in a controlled environment. In the scope of our work, we define all non-typical behavior as the key events to be integrated in a mobility dataset. Contemporary mobility frameworks are able to create synthetic mobility data, but we identified a limited support to integrate customized key events. Thus, we propose in this work a methodology to address this gap.

The rest of this paper is structured as followed. In the subsequent Section 2 we discuss the current state of the art in modeling human mobility and the associated activity schedulers. Further, we review in Section 2.2 on previous work done on intra-day activity schedulers and discuss strengths and weaknesses of available mobility frameworks regarding the injection of preset activities and identify a research gap on integrating controlled key events. Based on the review of related work we suggest a methodology in Section 3 and thus our attempt in closing this gap. We validate our method by providing a possible implementation in Section 4.1 using Python 3.10 and validate against predefined requirements from Section 3.3. Finally, we discuss our results and derive possible future work in Section 5.

2 RELATED WORK

In this section we present related work to our approach. We screen the online databases of the four largest IT publishers *IEEE Xplore*, *ACM Digital Library*, *Elsevier Science Direct* and *Springer Link* and two scientific search engines *Google Scholar* and *ResearchGate* between years 2010 and 2022. Our result is that activity-based models are often designed in integrated systems, hence a data scientist may provide variable initial settings, but the systems feature limited means to inject static activities (i.e. the key events) in the activity planning phase at the agent's individual level.

2.1 Modeling Human Mobility and Activity Schedulers

One intuitive approach to model human mobility of single individuals is the use of activity-based models as a bottom-up approach for simulating human decisions in a resource-limited environment (Castiglione et al., 2015). According to Zheng et al. (Zheng et al., 2013) agent-based models are often rooted on activity-based models, where agents are autonomous individuals that may include their past experiences in future decisions, e.g., using reinforced learning algorithms. Agent-based models are a common choice within the research community to simulate human mobility of small groups up to complete populations concurring in a joint environment with limited resources, e.g., temporal and spatial constraints.

The models are often implemented in integrated systems, i.e. the system gradually adapts to changes without user intervention. For instance, the authors Luca et al. identify in their work (Luca et al., 2021) missing control in solutions based on deep learning. Thus, the systems offer limited means to induce controlled activities and therefore force agents' behaviors to be adapted accordingly. Further, these models are data-driven and rely on high amounts of representative data obtained from the target region matching the desired use case.

The review on human mobility models by Solmaz and Turgut in 2019 (Solmaz and Turgut, 2019) determined a need in creating scenario-specific and realistic human mobility models. According to their work, available intra-day mobility models mainly focus on creating typical behavior, e.g., in a sleep-work-leisure cycle as in the Working Day Movement Model (Ekman et al., 2008). Modeling non-typical behavior, i.e. inducing scenario-specific activities in available models, are currently not in focus of current research, although Ekman et al. (Ekman et al., 2008) determined a general research trend towards scenario-specific models. These models might be used to generate mobility datasets simulating the usage of a wide range of devices and vehicles, according to the current mode of transportation.

Activity schedulers are used to plan and execute activities from a set of possible activities, where activities represent the coarse choices of an individual. This can be performed on a daily basis according to the individual's current needs and desires. For instance, if one individual's need is to buy groceries, then an activity scheduler would search for grocery stores in the near vicinity according to his desires. In most cases, humans start using different modes of transportation to their destination, and again according to their desires. After choosing an activity, a scheduler would then calculate the estimated time slot of the activity and executing it, considering any changes in the environment, e.g. traffic jams may affect the travel time or alternate faster routes may be the better choice. In most mobility frameworks, an intra-day activity scheduler is often combined using a microscopic mobility models to generate the tracking data at high spatial and temporal resolutions, e.g. applying routing mechanisms in a vehicular road network (Castiglione et al., 2015).

2.2 Agent-Based Systems

An example of an agent-based framework is *SimMobility* created by Adnan et al. (Adnan et al., 2015) using multiple modes of transportation. This framework is integrated and consists of different sub-models to simulate a range of complex context-aware decisions and mobility at different time steps in a demand-and-supply paradigm. Changes in one of the sub-models are propagated to the others sub-models. Due to it's system design, it is not intended to adapt the scheduling process to integrate further preset activities.

The open-source project MATSim (Horni et al., 2016) is an agent-based modeling framework to efficiently simulate a wide range of use cases of intra-day activities, using different modes of transportation. The scheduler employ a set of different strategies in a optimization problem setting, even when the initial data is not complete, as agents automatically adapt to find better solutions. Due to it's design, it provides a method to interfere with the daily activity planning, but as induced activities might not provide an optimal solution it may not be selected and is not suitable including preset activities in the scheduling process.

Another tool is the work by Bösche et al. (Bösche et al., 2012). Their tool creates synthetic data based on statistical properties of origin-destination pairs on travel surveys. While their synthetic approach resembles the statistical properties of real data, it does not provide means to inject preset data, individual behaviors and event-based delays, e.g., roadblocks.

The authors Drchal et al. introduced in their work (Drchal et al., 2019) an activity-based scheduler applying various strategies to select and plan a set of intra-day activities, similar to the mid-term model of the SimMobility (Lu et al., 2015) framework. In detail, the authors (Drchal et al., 2019) select a set of possible next activities, calculate the estimate duration and travel time. From the given set of activities they choose the one with the highest attractiveness, according to the statistical properties of real data. Although, the scheduler needs real data to train the model, it can be used to generate additional data outside the training area, supposing that the sub-area is an adequate representative of the complete simulated area.

The authors Al-Kuwari and Wolthusen describe in their work (Al-Kuwari and Wolthusen, 2010) that the geospatial forensic analysis of confiscated devices rarely have a complete tracking dataset available and forensic experts need to manually reconstruct the tracks. Therefore, they introduce a multi-modal trace reconstruction algorithm by applying different modes of transportation, i.e. mobility models and reconstructing data using a probabilistic approach. While this approach focuses on modeling mobility between two positions, it does not provide means to integrate additional activities.

In our review we could not find any method to adapt the scheduling process for mobility activity-based models and thus integrating preset activities in the activity planning phase. Thus we suggest a possible approach in the following Section 3 where we attempt to close this research gap.

3 METHODOLOGY

We suggest in the following a method to extend available mobility models to integrate preset activities in the generation process. We contribute to close the research gap and propose a method which integrates a set of static key events and models additional synthetic events around them. The resulting set of preset and generated activities for each agent is then the basis for generating synthetic human mobility at a microscopic level.

In detail, we discuss the scheduling order in Section 3.1 using a hierarchical approach. Next, we describe the scheduler using the intermediate activity model to iteratively generate further activities in free time slots in Section 3.2. In Section 3.3 we define the attributes and constraints of activities needed to be considered for the intermediate activity model. Finally, we discuss how an agent's individual preference might be considered in scheduling additional activities in Section 3.4

3.1 Model Activity Order

Activities describe coarse tasks, which differ in the aspects *when*, *where* and *how* they can be executed and therefore have several constraints, i.e. how flexible they are in the scheduling process. The constraints can be described in time (e.g., working and service hours), space (e.g., swimming activities can only be performed at public swimming pools, at home or at a public lake) and type (e.g., transport activities highly depend on the available modes of transport). A random scheduler may produce activity plans that could exclude vital activities, e.g., work activities can be primary planned during working hours at predefined days of the week.

Therefore, we propose that activities should be scheduled according to their importance and/or by the degree of constraints, similar as in the ILUTE Model (Salvini and Miller, 2005). To simplify the scheduling, we use a numerical scoring to aggregate different constraints. Thus, activities with more constraints are scheduled with higher priority than activities with fewer constraints. Using preset activities as key events in activity plans are non-optional and should always be scheduled first, that is before adding other activities. This concept can also be further expanded to adjust to customized scheduling strategies and adapting the scoring, e.g., prioritizing activities in certain situations.

3.2 Intermediate Activity Model

To include preset activities, we adapt the scheduling process in using higher-constrained activities and schedule other lower-constrained activities in between, similar as in the intermediate stop location model (Castiglione et al., 2015) and name it accordingly: *intermediate activity model*. The intermediate stop model is originally used to simulate additional stops in a tour between two successive points. Thus, additional stops (or activities) are detours where the destination is usually a work place, school or a return home location.

Activity Time Windows. According to Castiglione et al. (Castiglione et al., 2015), one possibility to determine when activities should be scheduled, is the use of time windows. This time window specifies a time slot where additional activities and the corresponding trips might be generated. Preset activities in general might not start and end at fixed time periods and therefore a scheduler needs to dynamically adapt the time windows according to the actual starting and ending times of consecutive activities. When new activities are added to the activity plan, the scheduler adjusts the time window accordingly.

3.3 Attributes of Activities

Inserting additional activities between two successive activities according to the intermediate stop model forces a scheduler to be context-aware. This may cover the previous scheduled activities (e.g., reasoning chains), the mode of transport utilized in this tour (e.g., for mode chains modeling (Song et al., 2021)), the potential activities that might be chosen as intermediate stops in this time window (e.g., constrained by working and service hours), estimated travel times to reach the next destination, constraints placed on the environment (e.g., limiting the number of agents per area) and the agent's profile (i.e. his desires and needs). In the following, we describe the minimal necessary attributes for using the intermediate stop model inside a scenario.

Each activity A consists at least of a duration d, timestamps t at a starting point p_s and ending p_e point and an activity type class. Further we assume that each activity has the following constraints and attributes: Exclusivity, chain of reasoning, physical and time constrains, labels. In the following, we define the given constraints and attributes.

Exclusivity. At any time t, there is at most one activity assigned and therefore we can exclude that more than one activity is performed simultaneously, i.e. we can define an injective function $f : T \mapsto P$ with $t \in T$ and $A \subset P$, where T is the complete set of timestamps and P the activity plan containing all scheduled activities A. If an activity consists of a set of sub-activities, then these sub-activities should consider the temporal and spatial boundaries of it's parent, i.e. starting and ending times and positions. With this definition, we can order all available activities according to their starting times (or ending times) and conclude that two consecutive activities A_i and A_{i+1} do not contain any additional activities in between or overlapping each other. Based on the previous definitions, we can now insert additional activities in the time window between two consecutive activities as depicted in Figure 1, i.e. after the activity A_i and before the activity A_{i+1}.

Chain of Reasoning. The integration of new activities in a given time slot may induce several problems. Any new activity A_{new}, that is not in the close vicinity of the previous activity, may induce additional activities, e.g., creating a transport activity (I and III in Figure 1) according to the available modes of transport to reach a destination. Any newly added activity should contain sufficient time for an agent to reach the consecutive activity with any available modes of transportation.

Physical and Time Constraints. Human mobility in general is not without limitation. According to

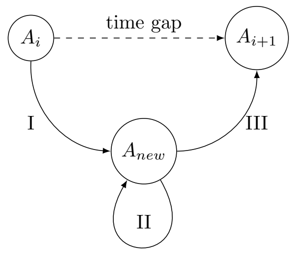

Figure 1: Intermediate tour model: A new activity A_{new} can be inserted between the given activities A_i and A_{i+1} having a fixed time gap. The scheduler needs to consider potential transport activities from A_i to A_{new} (I) and from A_{new} to A_{i+1} (III). The step in II can be repeated to recursively add more activities until either the time gap or the agent's needs are exhausted.

the mode of transportation or the activity, some areas or paths might not be accessible for specific time slots. For instance, shopping centers may have time restrictions and individuals are not permitted to enter this area outside service hours. Further, the choice of transport also influences variables like travel time (e.g., limited by speed limits), route choices (e.g., choosing routing alternatives during a traffic congestion) and path choices (e.g., in general vehicles travel on a directional network). This rule set should be extended to match the desired use case and is therefore not complete.

Labels. Any activity contains a description (e.g., a label) about the type of activity. This is not only needed by an algorithm to properly recognize it's type, but also include some level of documentation for the data scientist to reproduce the results. Appropriate descriptions might influence how, where and when activities are planned and executed. For instance, according to the chain of reasoning, some activities are dependent on previous decisions and labels aid to properly identify them.

3.4 Modeling Agent Preferences

Each agent has his own desires and needs, therefore we must consider that some agents might have preferences or restrictions in their choices of the activity type and mode of transportation. It depends, according to Castiglione et al. (Castiglione et al., 2015) on variables like income, social statuses, individual accessibility and availability to use different modes of transport and demographic data. This list should be extended to fit the current use case.

The agent's preferences might influence how, when and where activities are scheduled and thus limiting the choices for a scheduler. For instance, depending on the agent's occupation, we find different time windows when a certain work might be scheduled, e.g., night-shifts. Going further, we may identify, that the workplace is always performed at known location in contrast to maintenance works for the general public, where workers respond to house calls and constantly change their work place. Therefore, each activity type class should be modified to fit the agent's need in adding or changing the corresponding parameters, e.g., work hours differ for each agent, but are fixed.

4 VALIDATING THE METHODOLOGY

Although the suggested methodology might be scaled up to be used in a multi-agent system, it does not directly affect other agents. In addition, comparing the resulting dataset with real data or other models is limited, as key events might describe non-typical behavior. In this section we present a Python-based implementation and validate our methodology as presented in Section 3 to include preset activities for simulating a human mobility scenario.

4.1 Implementing the Activity Scheduler

The outline of the implementation approach of scheduling intra-day activity plans is depicted in Figure 2. It consists of a preparation phase, where the system cleans and transforms the input data to be processed in the next steps. During the next phase, the system synthesizes the population, i.e. we define for each agent their preferences and needs. In a daily cycle, the activities are hierarchically scheduled according to the model order as defined in Section 3.1. Each daily cycle ends in a simulation phase at the microscopical level by generating the actual tracks. Regular logical checks at different phases assure that scheduled activities are conform with the requirements as defined in Section 3.3. In the following we go into more detail for each of the implementation steps.

Activity Preprocessing Phase. In this preprocessing step, the scheduler identifies the type of the preset key events, the starting and ending locations and the timestamps at these locations. This phase is mainly associated with data cleaning and transforming user input data and creates activities to be processed in the next steps.

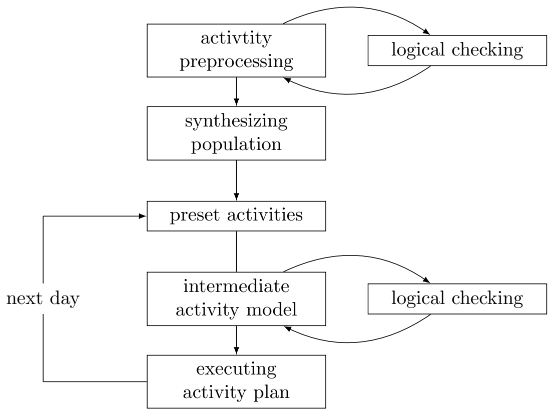

Figure 2: This chart describes the program flow of injecting preset activities. After preparing the preset data in a preprocessing phase, the population is synthesized in a consecutive step. In a daily cycle, the program schedules the preset activities and gradually plans additional activities according to their degree of constraints. Logical checks are executed at different phases to minimize logical discrepancies in the insertion of (preset) activities. In the last step, the activity plan is executed using microsimulation models

Synthesizing Population. In this intermediary step, we synthesize the population by creating agents, having their own desires and preferences. Here we define the needed global information like fixed locations (e.g., home position or workplace), when and how activities are scheduled by adjusting the initial settings for each agent. In this step, a data scientist can easily create different profiles for each agent.

Adding Preset Activities. On a daily basis, all preset activities are scheduled for the associated agents. Between two consecutive preset activities, we calculate the respective time window, where additional activities might be scheduled. As long as the time window is greater than a given threshold, we can iteratively plan new activities. The threshold is a combination of a minimum duration of an activity and the relative travel times between the activities, as described in the intermediate activity model.

Intermediate Activity Model. The intermediate activity model describes how additional activities are consecutively generated between two given activities. We implemented this model as described in Algorithm 1, where new activities are iteratively added. Given the two consecutive activities A_i and A_{i+1} and a potential new activity A_{new}, a scheduler generates a transport activity from the previously added activity A_{prev} to this new activity A_{new} and plans a transport activity from the newly generated activity to the last activity A_{i+1}. The new activity is only added if both travel times and the duration of the activity do not exceed the time window. The last transport activity is only of importance, if we can not create additional activities, i.e. the time window is too small for new activities to be added or there are global settings that do not permit creating more. In this case, we can execute the transport activity.

$A_{prev} \leftarrow A_i$;
while $\Delta_{gap} \leq \alpha$ **do**
  $generateTransport(A_{prev}, A_{new})$;
  $generateActivity(A_{new})$;
  $planTransport(A_{new}, A_{i+1})$;
  $\Delta_{gap} \leftarrow adaptTimeWindow()$;
  $A_{prev} \leftarrow A_{new}$;
end
$executeTransport(A_{prev}, A_{i+1})$;

Algorithm 1: Inserting intermediate activities A_{new} in a given time window Δ_{gap} of activities A_i and A_{i+1}.

Microsimulation Mobility Model. We used the python OSMNX library to download and extract the paths, points of interest and nodes of the OpenStreetMaps (OSM) database (Open Street Map, 2023). This framework also provides useful data cleaning methods, e.g., connecting street ends at intersections or calculating travel times between nodes. The concept of nodes are used in OSM to represent a multitude of things, but we are only interested in those representing connections in paths, e.g., intersections, curved ways or stop-points and edges represent subpaths, e.g., parts of a road. The routing of agents on a network is implemented as a weighted Dijkstra algorithm, where simulated travel times are the edges' weights between two nodes.

Logical Checking. To ensure that activities do not contain logical faults, we define a global set of rules prior generating new activities. The rule set can also be used to check for any human error prior integrating preset activities to the system during the preparation phase. In our system design, we included the following attributes from Section 3.3: *ordered set*, *exclusivity* and *chain of reasoning*.

As preset activities might describe non typical behavior, we did not include checks for preserving any time and physical constraints and individual preference, e.g., entering restricted areas after service hours. However, those attributes are considered as rules in the planning of new activities.

4.2 Validation

Therefore, to ensure that the implementation from Section 4.1 of our proposed methodology (see Section 3) provides the intended results, we validate the resulting dataset of one agent against the requirements from Section 3.3.

We provided our implementation with a set of fixed preset activities as our key events and generated additional activities around these activities. Considering that preset activities might represent any activity combinations of non-typical daily behavior, e.g., having a day off from work and performing other activities, we decided to randomize the preset dataset with a high variety of activity combinations. The methodology was implemented on a Windows 11 Pro (Version $21H2$) running on Intel Core i9-12900H with 64GB RAM using Python 3.8.11, OSMNX library version 1.1.1.

The Preset Activity Dataset. This dataset contains randomized activities, where the location during an activity does not change, i.e. the agent starts and ends the activity at the same location an no further tracking data is generated during his activity. Usually, the transition between two consecutive activities is modeled as a transport activity, given that the next activity is not at or in the near vicinity. We restricted our test data to be generated in a urban area of Munich, Germany with approximately $11km^2$. Further we adjusted the agent's profile, so that sleeping and at home activities are scheduled at the same location(, i.e. a physical constraint). To simulate a time constraint, we defined that sleeping activities can only be scheduled between 19:30 and 9:00 hours. The activities *shopping* and *work* activities should only be scheduled between 8:00 and 20:00 hours. The simulation order for the activities are: *work* as mandatory (highest scoring), *sleep* and *shopping* as restricted (medium scoring), *walk* and *at home* as unrestricted (lowest scoring).

Results. A sample intra-day activity plan containing preset and simulated activities is described in Table 1, where preset activities are labeled with the 'preset' tag in the simulation order column. Other labels, e.g., mandatory or restricted, describe the simulation order for this particular activity. Further, we identify that each activity complies with our requirements:

- **ordered set** all activities are ordered according to their starting/ending times.
- **exclusivity** at all times, we identify at most one single activity scheduled.
- **chain of reasoning** between two consecutive non-transport activities, there is always a transport activity, if not scheduled at the same location, which is always true in our sample.
- **physical constraints** all *at home* and *sleeping* activities are scheduled at the same location, i.e (48.18286, 11.52519).
- **time constraints** *sleeping* (entry 3) is scheduled between 22 and 9 hours and *shopping* (entry 13) is scheduled between 8 and 20 hours. We want to point out that the entry 11 is a preset activity and therefore not bound to this time constraint.

5 CONCLUSION

In this paper we identified the research gap, that the current state-of-the-art activity-based models on intra-day human mobility do not contain a possibility to adapt the scheduling process to include preset activities with fixed timestamps and locations. Having preset activities allows data scientists to injecting key events and thus provide a control mechanism in the generation of activity plans, which then may be used in a next step to generate customized geospatial data.

To close this gap, we proposed a methodology based on activity-based models to address this problem and defined the requirements for these preset activities, i.e. ordering, exclusivity, chain of reasoning and aware of time and physical constraints according to the agent's individual preferences. To validate our methodology we created a Python-based implementation using the *OSMNX* framework for integrating *OpenStreetMaps*. We purposed a hierarchical-based intermediary activity model to add new activities according to their degree of constraints, as described in Section 3.1. As preset activities might represent any combination of typical and non-typical behavior, we included a randomized set of preset activities and showed that the resulting set also complies with our requirements in Section 4.2.

Although our proposed methodology is able to integrate predefined activities in the scheduling process, it is limited to non-transport preset activities that start and end at the same location. For integrating predefined transport activities that contain different starting and ending positions, we may need to consider the reason behind this transport activity, i.e. what agent's need is met by this preset transport activity. This might possibly trigger implicit activities needing to be added and adapting the scheduling order, e.g., a transport activity between home and work location would implicate activities to be performed at home and at work. As the integration of transport activities is not trivial, we propose future work in further closing this research gap.

Table 1: A sample test dataset containing preset and simulated activities from one agent in a 24 hour window. The activities have physical and time constraints when and where they may be scheduled. Further, we identify that our requirements are met in the generated activities, i.e. ordered set, exclusivity and chain of reasoning as described in Section 3.3.

Entry	Sim. order	Activity type	Start Time	End Time	Start (lat, lon)	End (lat, lon)
1	preset	home	17:34:14	19:35:40	(48.18286, 11.52519)	(48.18286, 11.52519)
2	restricted	transport	19:35:40	19:35:40	(48.18286, 11.52519)	(48.18286, 11.52519)
3	restricted	sleep	19:35:40	02:53:24	(48.18286, 11.52519)	(48.18286, 11.52519)
4	unrestricted	transport	02:53:24	02:53:24	(48.18286, 11.52519)	(48.18286, 11.52519)
5	unrestricted	home	02:53:24	05:12:36	(48.18286, 11.52519)	(48.18286, 11.52519)
6	unrestricted	transport	05:12:36	05:13:59	(48.18286, 11.52519)	(48.18469, 11.53327)
7	unrestricted	walk	05:13:59	07:03:00	(48.18469, 11.53327)	(48.18469, 11.53327)
8	unrestricted	transport	07:03:00	07:08:47	(48.18469, 11.53327)	(48.18418, 11.49277)
9	unrestricted	walk	07:08:47	11:20:24	(48.18418, 11.49277)	(48.18418, 11.49277)
10	unrestricted	transport	11:20:24	11:25:34	(48.18418, 11.49277)	(48.18286, 11.52519)
11	preset	sleep	11:25:34	12:30:24	(48.18286, 11.52519)	(48.18286, 11.52519)
12	unrestricted	transport	12:30:24	12:34:36	(48.18286, 11.52519)	(48.18005, 11.49265)
13	unrestricted	shopping	12:34:36	13:15:54	(48.18005, 11.49265)	(48.18005, 11.49265)
14	unrestricted	transport	13:15:54	13:20:28	(48.18005, 11.49265)	(48.18286, 11.52519)
15	preset	home	13:20:28	17:16:49	(48.18286, 11.52519)	(48.18286, 11.52519)

REFERENCES

Adnan, M., Pereira, F. C., Azevedo, C. M. L., Basak, K., Lovric, M., Raveau, S., Zhu, Y., Ferreira, J., Zegras, C., and Ben-Akiva, M. (2015). Simmobility: A multi-scale integrated agent-based simulation platform. In *95th Annual Meeting of the Transportation Research Board Forthcoming in Transportation Research Record*.

Al-Kuwari, S. and Wolthusen, S. (2010). Forensic tracking and mobility prediction in vehicular networks. In *Advances in Digital Forensics VI*, pages 91–105. Springer Berlin Heidelberg.

Aschenbruck, N., Munjal, A., and Camp, T. (2010). Trace-based mobility modeling for multi-hop wireless networks. *Computer Communications*, 34(6):704–714.

Barbosa, H., Barthelemy, M., Ghoshal, G., James, C. R., Lenormand, M., Louail, T., Menezes, R., Ramasco, J. J., Simini, F., and Tomasini, M. (2018). Human mobility: Models and applications. *Physics Reports*, 734:1–74. Human mobility: Models and applications.

Bösche, K., Sellam, T., Pirk, H., Beier, R., Mieth, P., and Manegold, S. (2012). Scalable generation of synthetic GPS traces with real-life data characteristics. In *Technology Conference on Performance Evaluation and Benchmarking*, pages 140–155. Springer.

Castiglione, J., Bradley, M., and Gliebe, J. (2015). *Activity-Based Travel Demand Models: A Primer*. The National Academies Press, Washington, DC.

Drchal, J., Čertický, M., and Jakob, M. (2019). Data-driven activity scheduler for agent-based mobility models. *Transportation Research Part C: Emerging Technologies*, 98:370–390.

Ekman, F., Keränen, A., Karvo, J., and Ott, J. (2008). Working Day Movement Model. In *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, pages 33–40.

European Parliament and Council (2016). Regulation (eu) 2016/679. *Official Journal of the European Union*, vol. 59(L 119):1–88.

Holmes, R. and Holmes, S. (2008). *Profiling Violent Crimes: An Investigative Tool*. SAGE Publications.

Horni, A., Nagel, K., and Axhausen, K., editors (2016). *Multi-Agent Transport Simulation MATSim*. Ubiquity Press, London.

Lu, Y., Adnan, M., Basak, K., Pereira, F. C., Carrion, C., Saber, V. H., Loganathan, H., and Ben-Akiva, M. E. (2015). Simmobility mid-term simulator: A state of the art integrated agent based demand and supply model. In *94th Annual Meeting of the Transportation Research Board, Washington, DC*.

Luca, M., Barlacchi, G., Lepri, B., and Pappalardo, L. (2021). A survey on deep learning for human mobility. *ACM Computing Surveys*, 55(1):1–44.

Open Street Map (2023). Open street maps. https://www.openstreetmap.org/. Accessed: 2023-02-10.

Salvini, P. and Miller, E. J. (2005). ILUTE: An operational prototype of a comprehensive microsimulation model of urban systems. *Networks and Spatial Economics*, 5(2):217–234.

Solmaz, G. and Turgut, D. (2019). A survey of human mobility models. *IEEE Access*, 7:125711–125731.

Song, Y., Li, D., Cao, Q., Yang, M., and Ren, G. (2021). The whole day path planning problem incorporating mode chains modeling in the era of mobility as a service. *Transportation Research Part C: Emerging Technologies*, 132:103360.

Zheng, H., Son, Y.-J., Chiu, Y.-C., Head, L., Feng, Y., Xi, H., Kim, S., and Hickman, M. (2013). A Primer for Agent-Based Simulation and Modeling in Transportation Applications. Technical report, Federal Highway Administration.