Towards Automated Planning of Level Structures for Digital Interventions

Lorenzo J. James[1,3,*,†], Emanuele De Pellegrin[2,*,†], Laura Genga[1], Barbara Montagne[3], Pieter Van Gorp[1] and Ronald P. A. Petrick[2]

[1]Eindhoven Technical University, Groene Loper 3, 5612 AE, Eindhoven, The Netherlands
[2]Heriot-Watt University, Edinburgh Campus, Edinburgh, EH14 4AS, United Kingdom
[3]GGZ Centraal, Utrechtseweg 266, Amersfoort, 3818 EW, The Netherlands

Abstract

As global populations continue to age and chronic health conditions rise, the demand for scalable, effective rehabilitation interventions is increasing. Digital interventions offer a cost-effective solution for promoting healthy behavior change and self-management. However, traditional methods of handcrafting structured intervention content are time-inefficient, and not scalable. We propose a new approach to content structuring by modeling it as an automated planning problem. This work introduces the Automated Planning of LEvel Systems (APLES) tool, which leverages automated planning to generate structured intervention content within predefined constraints, balancing automation and designer control. APLES addresses the limitations of existing solutions such as Procedural Content Generation (PCG) systems, which often limit creative input. APLES allows designers to set rules that guide the automated processes. In this work-in-progress paper, we describe the technical progress of the APLES tool, including its planning framework and practical application in digital health interventions. Preliminary results demonstrate the tool's potential to improve efficiency and scalability in content structuring while maintaining alignment with creative goals. In future work, we will evaluate APLES across multiple digital interventions to assess its impact on user engagement compared to structures handcrafted by campaign managers, and its perceived usefulness by healthcare providers.

Keywords

Automated Planning, Levels, Level Generation, Gamification, AI, Health Intervention, Digital Intervention, Flow

1. Introduction

Worldwide, it is estimated that up to a third of the population has a health condition that could benefit from rehabilitation [2]. The need for rehabilitation is spread across the entire lifespan, from children with physical or intellectual challenges to adults with non-communicable diseases and elderly individuals with age-related difficulties [3, 2]. As populations continue to age, the number of individuals with chronic conditions and needs for rehabilitation will continue to grow [4, 2]. Through healthy behavior changes such as an increase in physical activity, healthier eating habits, less tobacco use, and more accessible and affordable rehabilitation interventions, population health outcomes can potentially improve [4, 5].

In the last few years, health interventions have become more accessible due to the significant increase in digital health interventions delivered through technologies such as smartphones, web applications, robotics, and wearable devices [5, 6]. Digital Interventions are a cost-effective way to promote healthy behaviors and self-management in users, compared to practitioner-delivered interventions [7]. Digital

AI4CC-IPS-RCRA-SPIRIT 2024: International Workshop on Artificial Intelligence for Climate Change, Italian Workshop on Planning and Scheduling, RCRA Workshop on Experimental evaluation of algorithms for solving problems with combinatorial explosion, and SPIRIT Workshop on Strategies, Prediction, Interaction, and Reasoning in Italy. November 25-28th, 2024, Bolzano, Italy [1].

*Corresponding author.
†These authors contributed equally.
l.j.james@tue.nl (L. J. James); ed50@hw.ac.uk (E. D. Pellegrin); l.genga@tue.nl (L. Genga); b.montagne@ggzcentraal.nl (B. Montagne); p.m.e.v.gorp@tue.nl (P. V. Gorp); r.petrick@hw.ac.uk (R. P. A. Petrick)
0009-0003-2505-8057 (L. J. James); 0000-0002-5979-6547 (E. D. Pellegrin); 0000-0001-8746-8826 (L. Genga); 0009-0006-8435-1407 (B. Montagne); 0000-0001-5197-3986 (P. V. Gorp); 0000-0002-3386-9568 (R. P. A. Petrick)

interventions often consist of applications that motivate participants toward health-related behavior change [8]. Within digital interventions, behavior change is promoted by facilitating healthy lifestyle activities, typically developed or approved in collaboration with experts in relevant health domains [9, 7]. The visualization of these activities is communicated through designed user interfaces, such as progress bars and activity feeds, which deliver information on activity completion and overall health goals [10, 11]. When designing digital interventions, the flow of information to the participants is handcrafted. This process involves the organizer of the digital intervention (i.e., campaign manager) manually designing a series of structured activities and tailoring the information flow to guide participants toward completing health-related goals effectively.

While handcrafting content structures in digital interventions are effective, they are also time-inefficient, labor-intensive, and not scalable for applications intended for large numbers of users [12]. This challenge highlights the need for more automated and flexible approaches to content creation and structuring. A common solution to the handcrafting design of digital applications is using Procedural Content Generation (PCG) [13]. PCG is the use of AI to create content for digital applications [14, 15]. In previous research, PCG has widely been used to generate content for applications such as characters and narrative using self-made algorithms [16], in-game challenges and messages using Genetic Algorithms [17], and activities using Large Language Models (LLMs) [18]. However, these systems limit creative control by automating much of the content creation process [14]. This trade-off between automation and control highlights the need for a system that can both automate and allow for more designer input. Such a system would enable designers to set rules for the automated processes to better align with creative goals and specific project requirements, such as controlling the order of activities. Structuring content within restrictions can be modeled as a state space problem [19]. This opens the possibility of applying automated planning [20], an AI technique that generates a sequence of actions to achieve a goal under a given set of constraints [21], in the largely unexplored context of structuring content for digital interventions. This approach could potentially offer a controlled yet automated method for structuring content in digital interventions.

In this paper, we introduce the Automated Planning of LEvel Systems tool (APLES). APLES is designed for digital campaign managers to create a structured series of activities based on rules. The APLES tool determines which activity will be visible at any given moment, according to rules set by the campaign manager. The types of rules that can currently be applied in APLES include determining when certain activities can be shown, the difficulty level of activities at different stages, and the overall difficulty of overtime progression. The tool also incorporates a flow chart that represents the difficulty curve users will experience, based on the principles of flow theory [22], ensuring the system adapts to and tests the user's skill level incrementally. Ultimately, APLES outputs a level structure that governs the progression of activities in the digital intervention campaign.

This work-in-progress paper outlines the development of the APLES tool. An evaluation of the tool with participants is currently planned but is out of the scope of this paper. The key contribution of this research is providing a practical tool for digital campaign managers: APLES automates the structuring of intervention activities and offers a foundation for further exploration in automated planning and level generation, with potential applications ranging from mHealth applications to rehabilitation robots.

2. Theoretical Background

2.1. Automated Planning

Automated Planning is a task that involves reasoning about generating a sequence of actions (plan) that achieves a set of predefined goals [20, 23]. A planning problem Π can be modeled as a tuple $\Pi = \langle P, A, S, I, G \rangle$, where P is a set of propositions or numerical variables (i.e., collectively known as fluents) that define a state space (i.e., including possibly a set of objects), A is a set of instantaneous actions that modify the value of the fluents when executed, S is a set of states where each state is an assignment of values to all fluents, $I \in S$ is a set of initial state properties, and G is the set of goal conditions to be achieved. The conditions in G are expressed in terms of the propositions and variables in P. Specifically, G defines the desired values of certain fluents from P that the planner needs to satisfy. Every action $a \in A$ is formed by a set of preconditions pre_a and a set of effects eff_a. Preconditions pre_a are logical expressions formulated using the fluents defined in P. They specify the necessary conditions that must hold in a given state s (or more formally be a subset of s) for the action to be applicable. The effects eff_a describe how the application of action a modifies a state s into a new state s'. They are formulated using the fluents in P and consist of assignments or updates of the values of these fluents. If pre_a holds in s then the application of action A results in the state change according to eff_a. A solution to the planning problem is a sequence of actions, called a plan, that when applied to the initial state transforms I to a state in which the goal conditions G are true.

Although the application of automated planning in the context of digital interventions is largely unexplored, previous research has used planning to optimize activity difficulty and personalize interventions. Vemuri et al. [24] introduced a multi-agent architecture for health coaching that dynamically adjusts goal difficulty based on real-time participant behavior, allowing for personalized goal selection tailored to individual user preferences in mHealth applications. Similarly, Pirolli et al. [25] applied user modeling and planning to improve self-efficacy and goal adherence, particularly in mHealth, by selecting personalized physical behavior goals to support user success.

2.2. Digital Interventions

Digital interventions are programs and devices that use digital technology to promote healthy behaviors and user self-management, allowing users to independently work on activities that are beneficial to them, with minimal to no direct involvement from healthcare professionals [7]. The positive impact of digital interventions on healthy behaviors is especially effective when grounded in behavior change theory [26]. Behavior change theories provide a scientific framework for effectively promoting healthy behaviors. According to self-determination theory [27], motivation is a key factor in driving behavior change [27]. Intrinsically motivated goals tend to foster higher engagement and persistence, leading to more sustained behavior change [27]. According to the goal-setting theory, setting specific yet challenging goals improves the motivation to achieve them. For a goal to be specific, there are five determinants: *clarity* (i.e., how clear is it), *challenge* (i.e., how difficult is it), *commitment* (i.e., how committed is the user), *feedback* (i.e., how well does the user see their progression), and *task complexity* (i.e., how complex are the subtasks of the goal) [28]. The APLES tool requires the campaign manager to input small, clear, and specific goals into the system and specify the difficulty of each goal.

The campaign manager can ensure that the difficulty curve of the activities selected by the APLES tool keeps the users of the intervention in *flow*. Flow is a mental state that makes an individual fully immersed in the activity they are currently performing [22]. Users enter the flow state when the activities they are performing are neither too difficult nor too easy for their current skill level. When an activity is too difficult for their skill level users become frustrated. When an activity is too easy for their skill level users become bored with the activity [22]. Digital applications provide a dynamic interplay of challenge/difficulty and interleaving of different activities users can perform to keep players in flow, resulting in more engagement [29]. Flow is particularly promoted in digital applications designed using elements of game design, also known as gamified applications. A particular gamification element used to promote flow is the elements of level systems, as they match the activities available to users to their current skill level [30]. Level systems are properly planned sequences of events. The intensity of the events is structured with peaks and troughs, with a pacing similar to a three-act movie structure [31]. Level structures can be formulated in a planning task, that is solved using the APLES tool. Additionally, campaign managers can easily visualize these level structures using a graph within the APLES tool, potentially making it easier to control the difficulty curve.

3. The APLES System

The architecture of the APLES system was designed to create a modular and scalable tool for level structure generation. A planning library was employed to define and solve the planning problems.

Listing 1: Generated action example using UP in PDDL format. Showing also goal definition and metric.

```
(:functions
    (difficulty_lvl ?d - difficulty)
    (difficulty_lvl_physical ?d - difficulty)
    (difficulty_lvl_social ?d - difficulty)
    (cost_take_a_15_minute_walk)
    (cost_run_5_km)
    (total-cost))

(:action take_a_15_minute_walk
    :parameters ( ?d - difficulty ?atype - physical_0)
    :precondition (and (candoactivitytype ?atype))
    :effect (and
        (increase (difficulty_lvl_physical ?d) 2)
        (increase (cost_take_a_15_minute_walk) 1)))

(:action run_5_km
    :parameters ( ?d - difficulty ?atype - physical_0)
    :precondition (and (candoactivitytype ?atype))
    :effect (and
        (increase (difficulty_lvl_physical ?d) 2)
        (increase (cost_run_500_km_) 1)))


(:goal (and
    (=(difficulty_lvl_physical counter) 4)
    (=(difficulty_lvl_social counter) 5)
    (=(difficulty_lvl_cognitive counter) 3)))

(:metric minimize
    (total-cost)
    (=(cost_take_a_15_minute_walk)(+(cost_take_a_15_minute_walk)1)
    (=(cost_run_5_km)(+(cost_run_500_km) 1))
```

Additionally, APLES was integrated with a digital intervention tool to demonstrate its application, and with a plan visualization tool to assist users in visualizing the planning process.

3.1. Unified Planning

Unified Planning (UP) is a Python library provided by the AIPlan4EU project[1] with the aim of simplifying the use of automated planning tools for AI application development by providing a planner-agnostic method for defining planning problems. UP attempts to standardize aspects of the planning process, making it accessible to users of any level of expertise. In particular, it offers a well-developed PDDL [32] parser a standard interface for communicating with external planners, and common operations such as grounding and validation. An example of the planning model generated by APLES is shown in Figure 1, representing the functions, actions goal, and metrics definitions. UP is very well positioned for the type of project presented in the paper that requires a dynamic restructuring of the domain definition. In particular, UP is used to dynamically add functions, actions, and metrics based on the activity set-up in the front end.

[1]https://www.aiplan4eu-project.eu/

3.1.1. Problem Representation

The planning actions in level generation for a health intervention are the representations of various activities. Each action is associated with a specific difficulty score, which serves as a metric for assessing the intensity of the activity being in the 3 activity types: physical, social, or cognitive. The primary objective is to generate a plan that corresponds to an overall difficulty score in a particular activity type and ensure that the selected activities are appropriately challenging. The activities generated by the planner need to be engaging and conducive to sustained participation, therefore they should adhere to principles of flow theory, showing progress that keeps the user engaged avoiding frustration or boredom. To clarify, we'll be using the terms activities and actions interchangeably as the user activities are represented by actions in the planning problem.

Some actions or activities may require preliminary preparation before execution, such as engaging with video tutorials or other instructional materials. A critical constraint of the model is that actions should not be repeated, if possible, or keep the same activity repetition low maintaining variety and preventing monotony in the user's experience. The planning process assumes an initial starting point where no prior progress has been made before the execution of the plan. All actions have an associated cost function. For example, the activity "take a 15-minute walk" has a cost function called $cost\_take\_15\_min\_walk$ that increases the overall cost of the action when the planner chooses this action. The planner uses these costs in its quality metrics to optimize the plan by minimizing the cumulative cost of the activities it selects. The main idea behind representing activities with cost is to enforce APLES to not select the same action multiple times in a level.

The goal of the planning process is to reach a target difficulty level within a specific type of activity, aligning with the user's health objectives. To achieve all the proposed objectives the decision was to use numerical planning to represent parts of the problem such as the goal that is represented by a target integer number for each activity: $G := (4 \leq difficulty\_lvl\_physical)(3 \leq difficulty\_lvl\_social)(2 \leq difficulty\_lvl\_cognitive)$. The main numerical planner used for the development of APLES is ENHSP[33], as UP supports the optimal solution feature of this planner.

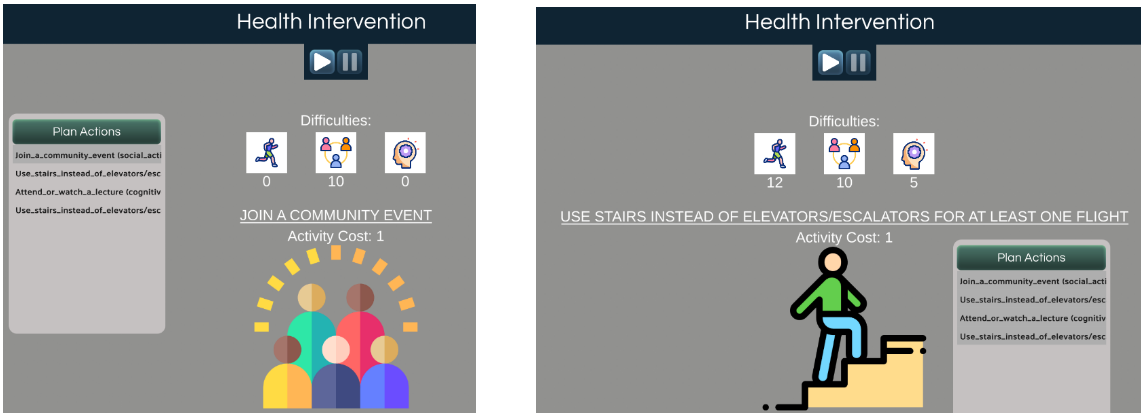
3.2. PDSim

PDSim (Planning Domain Simulation) [34] is a framework designed to facilitate the visualization of planning problems by integrating the representation of planning models with interactive 3D or 2D environments. The system leverages the capabilities of the Unity game engine[2], providing a platform for animating and interacting with planning problems. PDSim is particularly useful for understanding the dynamics of planning problems using visual cues rather than basic text output formats. PDSim translates planning actions and predicates into animations, providing a clear representation of the plan execution. This visual feedback helps in understanding the sequence of actions and the changes in the state of the world. PDSim is also capable of handling numerical planning problems, where actions have quantitative effects (i.e., resource consumption or temporal constraints). This is particularly useful in visualizing the planning problem defined in this paper as shown in Figure 1 representing a level in APLES. PDSim visualizes the changes in numerical values, such as keeping track of the difficulty of a level, and the increasing action costs, alongside the execution of actions. The system's ability to animate these numerical changes provides an intuitive understanding of how quantitative factors influence the overall plan.

3.3. GameBus

GameBus is a gamification engine enabling researchers to create configurations of health applications for use in health interventions [35]. In this study, we used a custom GameBus configuration in the form of an mHealth application as our digital health intervention application. In the GameBus application users can self-report the completion of healthy activities on the task page of the application. When

[2]https://unity.com/



(a) Visualisation of the first action in the plan.

(b) Final visualisation for the level.

Figure 1: Plan visualization using PDSim to visualize a level generated with APLES.

self-reporting, users can optionally provide written descriptions, or photographic or video evidence as proof of completing the activity. Each activity in the application has points, and users can earn those points by completing the activity. In addition to the self-reported activities, GameBus allows third-party health applications (i.e., GoogleFit, H5P) to send objectively tracked data such as steps, aggregation of steps, and interactions with video content.

Our configuration of GameBus showed activities available to the user displayed in the form of levels. Each level within the application has a number (n) of activities. Levels get increasingly difficult as a user progresses through them. Once users have completed all activities within a level or scored enough points by completing enough activities within the level, the user advances to the next level. If the user does not complete a level within seven days, they will stay at their current level for the next seven days.

Configurations of health interventions made within the GameBus engine are stored as campaigns. These campaigns can be downloaded, edited, and saved as a Microsoft Excel file (.xlxs). The GameBus also supports uploading these .xlxs configurations through its publicly available API.

3.4. Architecture

We developed a prototype of the APLES tool that generates level structures for digital interventions. The architecture of APLES, and its integration with GameBus as a digital intervention tool and PDsim as a planning visualization tool, can be seen in Figure 2.

Interface

The user interface of APLES is an online web application that campaign managers can use to create their level structures. In the user interface, campaign managers can add new activities, edit and delete existing activities, edit level difficulty graphs for each activity type, and create the level structure.

All activities considered by the planner are visible in the activity table in the user interface. In the backend, the activity table is saved as a CSV file with each activity element as a separate column. The Name column describes the activity, the difficulty describes the difficulty of the activity as an integer, Type describes the type of activity it is, CurrentCost indicates to the planner what the current weight of the activity is, CostIncrease indicates how much the weight of the activity will increase once the planner has selected it, Steps indicate how many steps a participant needs to perform in one session to complete this activity, and StepsAggregate indicates how many steps within a day a participant needs to do to complete this activity. Both steps and stepsAggregate decide how many steps need to objectively be tracked, or self-reported before the system deems the activity completed.

Campaign managers can configure the difficulty of the activities at each point of the intervention by editing the difficulty graph. In the difficulty graph, campaign managers can set the number of levels

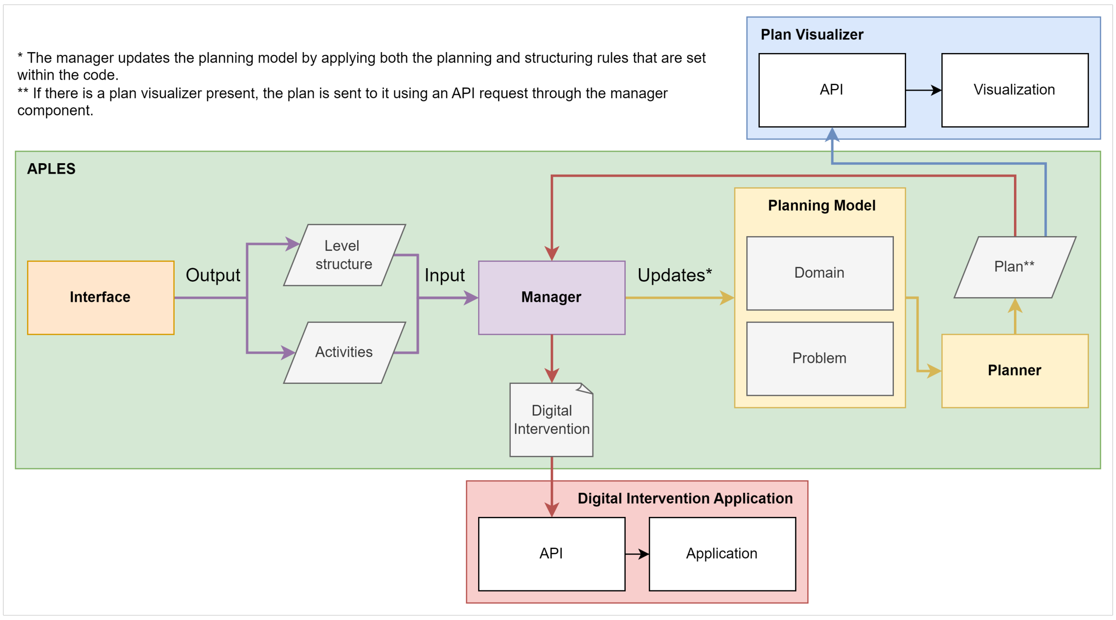


Figure 2: High-level architecture of APLES and its connection to PDsim and GameBus.

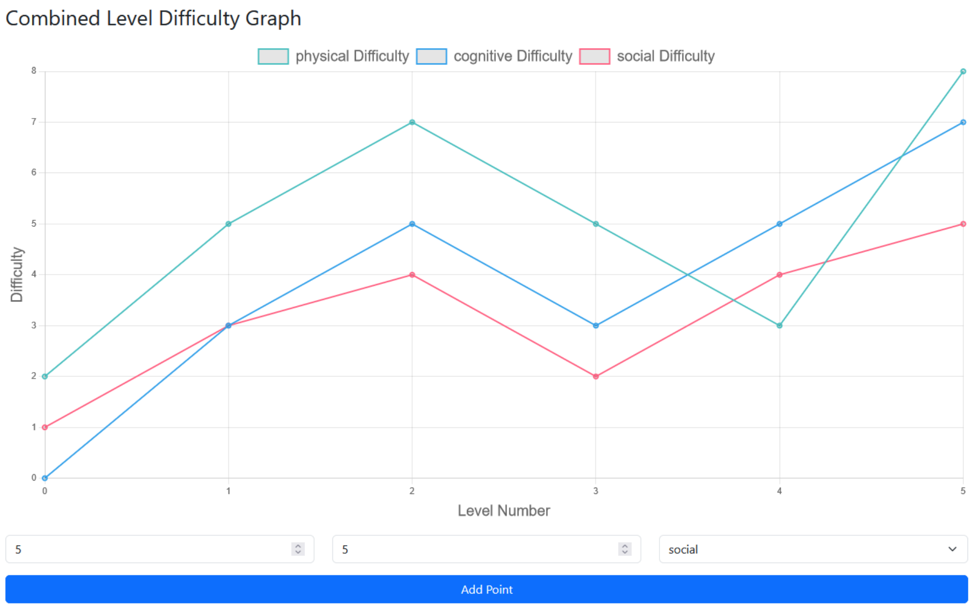


Figure 3: Level difficulty graph available in the interface of the APLES tool. The Y-axis shows the difficulty level, the X-axis shows the level, and the colors represent the activity types within a level. At the bottom is a form available to add points to the graph in real time.

they want in their intervention, the difficulty for each of the levels, and the difficulty of each type of activity present in the levels. The graph visually showcases the difficulty of each level in real-time as the campaign manager enters the fields, as seen in Figure 3. If the user does not want a type of activity to be present within a certain level, the user can keep the difficulty level of that activity at zero. The level difficulty graph displays levels on the X-axis and difficulty on the Y-axis. The visualization of the level difficulty graph is inspired by the Flow theory graph [22]. The output of the level difficulty graph is an array of data points that represent the difficulty level for each activity type, per level. This array is sent to the Manager component of the system to generate levels.

Table 1
Examples of Typical Activities in APLES

Activities	METScore	Type	Cost	CostIncrease
Take a 25' walk	6	Physical	1	2
Go to a Social Event	2	Social	2	1
Read a Book	3	Cognitive	0	1
Watch video of a Physical Activity	1	General	0	0
...	...	...	...	...

Manager

The manager component of the APLES tool is responsible for creating the planning model, and updating it dynamically after each plan is created. The planning model (i.e., the domain model and the problem) is generated based on the input given to the manager by the interface component. Campaign managers can add rules that the APLES manager component needs to follow when generating the domain model. An example of a rule implemented in the system is that when participants are given an activity type they have not encountered before, the planner should assign a $tutorial\_video$ activity associated with that activity type to be created. Once a plan has selected the action to generate a tutorial_video of that type, the manager then updates the fluent $can\_do\_activity\_type(\$activity\_type)$ and sets it to True. This updates the domain knowledge of the planner, indicating to the planner the participant can do this activity and does not need to complete the tutorial_video activity of that activity type anymore, resulting in the planner not selecting this action anymore. In the initialization phase of the application, the manager component sets the weight of each activity using the unified planning method $problem.add\_quality\_metric(MinimizeActionCosts())$. The weights are based on the current cost attribute assigned to the activity. Each time a new plan has been sent to the manager component, the manager component updates the weight of each activity, based on the cost increase assigned to that activity.

Once the domain has been generated, the manager component sends both the domain and the problem files to the planner. The planner then returns the plan to the manager. Once the manager receives a plan it uses the plan to generate a level. Each plan the manager receives gets mapped to a separate level, resulting in the complete level structure. Another type of rule supported by the APLES system is level structure rules. These are rules that apply when creating the level structure. An example of this rule is that tutorial_video activities should always be within their separate level. The manager component splits the activities from a planned plan created by the planner into separate levels.

In APLES, activities are represented as standard planning actions with an associated cost. Activities are user-defined typically by an expert and loaded in the manager. Table 1 shows an example of the format APLES use to load a problem. In particular, it shows the difficulty score of each activity that is used to formulate the final goal of a particular activity type, the current cost that is updated every cycle after a level is completed (i.e., plan executed), and the cost increase value for each activity (a treated activity cost is less prone to get selected again). Finally, if an action is not executed in a level it will decrease its cost thus the planner might be able to select it again for the next level.

The manager component is also responsible for formatting the levels and the activities within them to a format that is accepted by the digital intervention application that is being used. In our implementation, the manager formats the activities into GameBus activity types, formats the levels into GameBus levels, and exports this data into a .xlxs supported by the GameBus application. An example of the differences between the plans and the level system formatted and generated by APLES can be seen respectively in Table 2 and Table 3.

The manager component also supports formatting activities to allow external applications connected to the digital intervention application to access them. An example of this feature is implemented using the GameBus application. Outside of self-reporting activities within the GameBus application, GameBus supports data input from external applications such as H5P, Google Fit, and Strava. The manager

Table 2
Two plans generated by the planner

Plan 1	Plan 2
tutorial_video(cognitive_activity)	Learn_a_new_phrase(cognitive_activity)
tutorial_video(physical_activity)	Walk_10000_steps_in_a_day(physical_activity)
tutorial_video(social_activity)	Learn_a_new_word(cognitive_activity)
Practice_learning_a_new_skill(cognitive_activity)	Learn_a_new_phrase(cognitive_activity)
Learn_a_new_phrase(cognitive_activity)	
Do_10_squats(physical_activity)	
Engage_with_others(social_activity)	
Play_a_boardgame(social_activity)	

Table 3
Levels created by the APLES based on the level structure rules

Level	Name	GameBus_Type	Points
1	tutorial_video(cognitive_activity)	H5P_GENERAL	1
2	tutorial_video(physical_activity)	H5P_GENERAL	1
3	tutorial_video(social_activity)	H5P_GENERAL	1
4	Practice_learning_a_new_skill	GENERAL_ACTIVITY	2
4	Do_10_squats	GENERAL_ACTIVITY	6
4	Engage_with_others	GENERAL_ACTIVITY	4
4	Play_a_boardgame	GENERAL_ACTIVITY	1
5	Learn_a_new_phrase	GENERAL_ACTIVITY	1
5	Walk_10000_steps_in_a_day	DAY_AGGREGATE	10
5	Learn_a_new_word	GENERAL_ACTIVITY	1
5	Learn_a_new_phrase	GENERAL_ACTIVITY	1

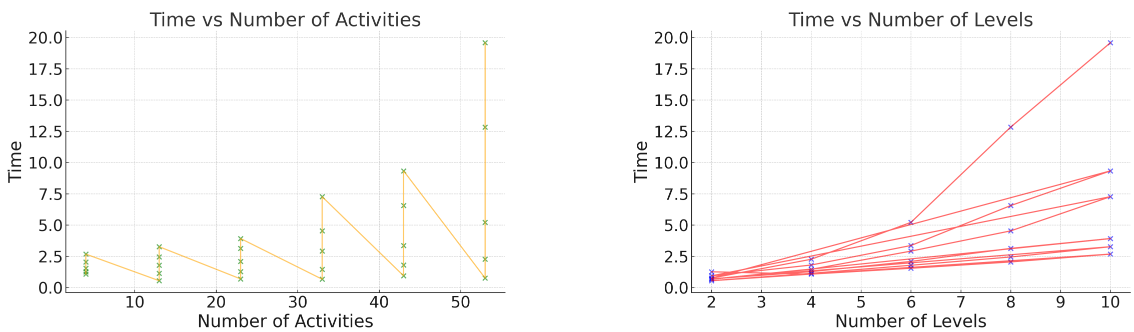
component of APLES ensures that activities that require walking/steps input supported by external applications are compatible with the activities. When an external application tracks the required steps to complete the activity created by the APLES system, the participant receives points within the GameBus application. The manager component also ensures that the tutorial videos are shown through H5P components in the GameBus application. Once the data is formatted, it sends the .xlxs file to GameBus through its API to generate the campaign fully compatible with GameBus and its supported external applications.

4. Results

APLES demonstrates the potential for scalability in particular if fixing the number of total levels to generate and increasing the number of total activities. To support this some level generation experiments have been performed. The test experiments were executed by increasing the total number of levels to generate(2 to 10) while increasing the number of total activities (4 to 53). The results are shown in Table 4 reporting the total time APLES take to generate a plan and generate levels. Results in Figures 4 show that if the number of levels increases (Figure 4b) the level generation time also increases with a possible non-linear pattern, likely indicating increasing computational complexity and performance degradation as the number of levels to generate grow. However, the level generation time seems more stable with respect to the increasing number of total activities in the system (Figure 4a), suggesting that APLES may handle this case more efficiently than the increase in the total number of levels.

Table 4
Time performances for level generation.

Total Number of Activities	Number of Physical Activities	Number of Social Activities	Number of Cognitive Activities	Number of General Activities	Total Number of Levels	Level Generation Time (s)
4	1	1	1	1	2	0.49
4	1	1	1	1	4	0.99
4	1	1	1	1	6	1.57
4	1	1	1	1	8	2.08
4	1	1	1	1	10	2.65
13	4	5	3	1	2	0.52
13	4	5	3	1	4	1.13
13	4	5	3	1	6	1.77
13	4	5	3	1	8	2.39
13	4	5	3	1	10	3.27
23	7	7	8	1	2	0.59
23	7	7	8	1	4	1.32
23	7	7	8	1	6	2.19
23	7	7	8	1	8	3.01
23	7	7	8	1	10	3.87
33	13	10	9	1	2	0.66
33	13	10	9	1	4	1.62
33	13	10	9	1	6	2.94
33	13	10	9	1	8	4.54
33	13	10	9	1	10	7.21
43	19	12	11	1	2	1.46
43	19	12	11	1	4	1.78
43	19	12	11	1	6	3.29
43	19	12	11	1	8	6.95
43	19	12	11	1	10	9.58
53	23	15	14	1	2	0.76
53	23	15	14	1	4	2.29
53	23	15	14	1	6	5.15
53	23	15	14	1	8	12.81
53	23	15	14	1	10	19.00



(a) Plot Level Generation Time vs Total Number of Activities.

(b) Plot Level Generation Time vs Total Number of Levels.

Figure 4: Level Generation Experiments Plots.

5. Discussion

5.1. Main Findings

This paper outlines the development of the APLES tool, a novel approach to automating structuring content in digital interventions. APLES aims to streamline the creation of levels or structured sequences of activities by leveraging automated planning techniques and applying the principles of Flow theory.

This approach enables the creation of activity sequences where the difficulty is dynamically adjusted to match the user's skill level, while also providing campaign managers with a customizable interface. Through this interface, managers can define specific rules regarding when activities should be presented, how difficulty should progress across levels, and how these parameters can be adjusted using a difficulty level graph. Campaign managers also control the activities the planner uses through the activity table. The campaign managers determine the difficulty and initial cost actions as part of the campaign through the activity table. This flexibility allows for more personalized interventions.

The campaign managers can input how campaigns should be structured, and APLES automatically creates campaigns based on the specified rules. This automation could potentially increase the time efficiency of campaign managers by removing the need to craft campaigns manually and giving them more time to focus on the structure and content of those campaigns.

The successful development of the APLES prototype demonstrates the feasibility of using automated planning for structuring content in digital interventions. Although the tool has yet to be evaluated in a live intervention, its initial implementation suggests it may help increase the scalability of digital health campaigns.

5.2. Limitations

While this project demonstrates the potential of using automated planning systems to create level structures for digital interventions, there are several limitations to the current implementation of the APLES tool.

A major limitation of the current APLES tool is the inability to dynamically add or remove both domain generation and level structure rules through the interface component. In the current system, adding and removing rules can only be achieved by modifying the code directly. This results in campaign managers who are not programmers not being able to easily adapt or extend the existing rule set that the planning component needs to take into account. The current rules supported by the system are confined to the rules presented within this paper.

Another limitation is that the current APLES tool does not support adding or removing activity types through the interface component. Currently, it does support directly adding activity types by modifying the underlying activities CSV file that populates the activities table. However, this file is saved on the server, hidden from campaign managers, and due to there being no functionality within the interface itself, this task may is not easy for the campaign managers.

There is currently, also a lack of support for multiple logins or databases resulting in all campaign managers using the same set of activities for their interventions. This design choice was due to the project's scope, however, the absence of multi-user support will pose challenges when multiple campaign managers use the APLES tool at the same time.

One of the planning limitations concerns the restriction associated with the exclusive use of numerical representations. Specifically, these limitations arise when activities do not consider the temporal aspects of the problem. For example, the current representation doesn't take into account the maximum or minimum duration of activities nor the explicit representation of deadlines or enforced start and end time frames.

Lastly, the user interface could benefit from improvements in user feedback and usability. Currently, when activities are added or removed, they are reflected in the activity table, but users do not receive any feedback indicating that these changes have been successfully completed. Additionally, when updating the graph, users must input every activity type, setting unwanted ones to zero to ensure the system functions correctly, potentially leading to frustration with the system.

5.3. Future work

In future work, the incorporation of temporal planning into the APLES tool should be explored [36]. By integrating temporal planning algorithms into the Manager component, the system could suggest activities based on temporal factors such as the time of year, week, or day. This would allow the system to offer more contextually relevant activity recommendations, such as suggesting intense outdoor activities during daylight hours or scheduling social events on weekends. The Interface component could also be updated to allow campaign managers to input specific temporal constraints, ensuring that activities are suggested at appropriate times. For temporal activities to be utilized effectively, the digital intervention tool should also support temporal activities, meaning that certain activities should appear based on the temporal conditions set by the APLES tool.

Another promising direction is exploring autonomous level structure graph generation using historical user data. Currently, the level structure graph requires manual input from campaign managers. Generating these graphs based on historical user generation could potentially personalize the difficulty curve of the level structure to the user's current skill level. Alternatively, the graph could also be generated based on rules provided by campaign managers, specifying the desired difficulty level. These options would reduce the need for campaign managers to manually create the graph and allow for a more adaptive and personalized experience for the participants.

A Human Digital Twin (HDT) is a virtual counterpart or part of a physical human, where the virtual twin receives information about the physical human (i.e., the physical twin) and can be used to predict elements of the physical human such as their behavior [37]. An HDT can be used to collect user data from multiple sources and can provide external contextual data to the planning model for personalizing the generation of levels. An example of this could be using the data to adjust the costs of the activities used in the planner based on user preferences and behavior patterns. The behavior patterns detected by an HDT could also potentially determine and possibly simulate if the difficulty graph is too difficult or too easy for a participant. The data provided by an HDT could potentially be used to dynamically generate and update personalized level structures.

Integrating LLMs could potentially be used to generate activities either from scratch or by expanding upon the existing activities within the APLES tool's activity table. By generating the activities, LLMs could automate much of the activity creation process, providing campaign managers with a starting point and preventing campaign managers from starting from scratch creating new activities for each intervention. Previous research has already explored the potential of using LLMs to generate activities that align with the SMART criteria by using a rubric, ensuring that the generated goals are measurable [18].

Finally, future work will involve evaluating the APLES tool in two different digital interventions across various contexts. The first intervention will use APLES' integration with the GameBus gamification engine to assess the impact of AI-generated level structures on participant engagement in a digital health intervention. A four-week, two-arm experimental trial is planned, where one group will receive a digital intervention with a human-created level structure, and the other will receive an APLES-generated level structure. Participants will engage with the application throughout the intervention period, and engagement will be measured subjectively using the Intrinsic Motivation Inventory (IMI) survey [38] and objectively by tracking their activity within GameBus.

The second intervention will use a rehabilitation robot in a healthy living lab setting. Each participant will interact with the robot during one 30 to 60-minute session. The campaign manager will use the APLES interface to design a series of activities to support the participant's rehabilitation. After the intervention, participants will complete HRI user experience forms. The interactions between participants and the robot and the use of the APLES tool will be recorded. Finally, healthcare professionals will evaluate the video recordings to assess the perceived usefulness of the APLES tool.

6. Conclusion

In this study, we present the development of the automated planning of LEvel Systems (APLES) tool. APLES is designed as a tool to help campaign managers automatically structure level systems for digital interventions across various contexts, ranging from mHealth interventions to robotics. By automating the structuring of intervention activities, APLES addresses the need for more time-efficient and scalable content structures in digital interventions. This paper outlines the architecture, implementation, and potential uses of the APLES system. However, the tool has not yet been evaluated by participants or campaign managers at this stage of the development. Future research will focus on evaluating the tool in digital interventions in the domains of health interventions and robotics, and refining the tool based on participant feedback. Ultimately, APLES represents a promising exploration of automated planning systems for planning health interventions.

Acknowledgments

Emanuele De Pellegrin is supported by EPSRC Centre for Doctoral Training in Robotics and Autonomous Systems (EP/S023208/1) at Heriot-Watt University and the University of Edinburgh. The authors would like to thank the GameBus team with a special thanks to Raoul Nuijten for bringing this paper together.

References

[1] D. Aineto, R. De Benedictis, M. Maratea, M. Mittelmann, G. Monaco, E. Scala, L. Serafini, I. Serina, F. Spegni, E. Tosello, A. Umbrico, M. Vallati (Eds.), Proceedings of the International Workshop on Artificial Intelligence for Climate Change, the Italian workshop on Planning and Scheduling, the RCRA Workshop on Experimental evaluation of algorithms for solving problems with combinatorial explosion, and the Workshop on Strategies, Prediction, Interaction, and Reasoning in Italy (AI4CC-IPS-RCRA-SPIRIT 2024), co-located with 23rd International Conference of the Italian Association for Artificial Intelligence (AIxIA 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[2] W. H. Organization, Rehabilitation 2030, https://www.who.int/initiatives/rehabilitation-2030, 2017. Accessed: 2024-09-03.

[3] G. Stucki, J. Bickenbach, C. Gutenbrunner, J. L. Melvin, Rehabilitation: The health strategy of the 21st century. (2018).

[4] W. H. Organization, Noncommunicable diseases, 2023. URL: https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases, accessed: 2024-09-03.

[5] L. Pearce, N. Costa, C. Sherrington, L. Hassett, Implementation of digital health interventions in rehabilitation: A scoping review, Clinical Rehabilitation 37 (2023) 1533–1551.

[6] M. Gomes, E. Murray, J. Raftery, Economic evaluation of digital health interventions: methodological issues and recommendations for practice, Pharmacoeconomics 40 (2022) 367–378.

[7] K. Bradbury, S. Watts, E. Arden-Close, L. Yardley, G. Lewith, Developing digital interventions: a methodological guide, Evidence-Based Complementary and Alternative Medicine 2014 (2014) 561320.

[8] R. Band, K. Bradbury, K. Morton, C. May, S. Michie, F. S. Mair, E. Murray, R. J. McManus, P. Little, L. Yardley, Intervention planning for a digital intervention for self-management of hypertension: a theory-, evidence-and person-based approach, Implementation Science 12 (2017) 1–13.

[9] S. M. White, E. L. Mailey, E. McAuley, Leading a physically active lifestyle: Effective individual behavior change strategies, ACSM's Health & Fitness Journal 14 (2010) 8–15.

[10] L. Xu, H. Shi, M. Shen, Y. Ni, X. Zhang, Y. Pang, T. Yu, X. Lian, T. Yu, X. Yang, et al., The effects of mhealth-based gamification interventions on participation in physical activity: systematic review, JMIR mHealth and uHealth 10 (2022) e27794.

[11] S. M. Phillips, K. S. Courneya, W. A. Welch, K. L. Gavin, A. Cottrell, A. Nielsen, P. Solk, D. Blanch-Hartigan, D. Cella, R. T. Ackermann, et al., Breast cancer survivors' preferences for mhealth physical activity interventions: findings from a mixed methods study, Journal of Cancer Survivorship 13 (2019) 292–305.
[12] S. Hornstein, K. Zantvoort, U. Lueken, B. Funk, K. Hilbert, Personalization strategies in digital mental health interventions: a systematic review and conceptual framework for depressive symptoms, Frontiers in digital health 5 (2023) 1170002.
[13] R. Van Der Linden, R. Lopes, R. Bidarra, Procedural generation of dungeons, IEEE Transactions on Computational Intelligence and AI in Games 6 (2013) 78–89.
[14] M. Hendrikx, S. Meijer, J. Van Der Velden, A. Iosup, Procedural content generation for games: A survey, ACM Trans. Multimedia Comput. Commun. Appl. 9 (2013). URL: https://doi.org/10.1145/2422956.2422957. doi:10.1145/2422956.2422957.
[15] N. Shaker, J. Togelius, M. J. Nelson, Procedural content generation in games (2016).
[16] Y. H. Pereira, R. Ueda, L. B. Galhardi, J. D. Brancher, Using procedural content generation for storytelling in a serious game called orange care, in: 2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), 2019, pp. 192–197. doi:10.1109/SBGames.2019.00033.
[17] E. Kalafatis, K. Mitsis, K. Zarkogianni, M. Athanasiou, A. Voutetakis, N. Nicolaides, E. Chatzidaki, N. Polychronaki, V. Chioti, P. Pervanidou, et al., Artificial intelligence based procedural content generation in serious games for health: The case of childhood obesity, in: International Conference on Wireless Mobile Communication and Healthcare, Springer, 2022, pp. 207–219.
[18] L. James, L. Genga, B. Montagne, M. Hagenaars, P. Van Gorp, Toward scalable content generation for gamified mhealth interventions: The evaluation of llm-generated goals on user engagement, 2024. URL: https://icec2024.uea.edu.br/index.html, 23rd IFIP International Conference on Entertainment Computing, ICEC 24 ; Conference date: 30-09-2024 Through 03-10-2024.
[19] L. James, Generating video game puzzles through planning, in: UK PlanSIG 2021 (ONLINE): The second online Workshop of the UK Planning & Scheduling Special Interest Group, 2021.
[20] M. Ghallab, D. Nau, P. Traverso, Automated Planning: theory and practice, Elsevier, 2004.
[21] S. J. Russell, P. Norvig, Artificial intelligence: a modern approach, Pearson, 2016.
[22] M. Csikszentmihalyi, M. Csikzentmihaly, Flow: The psychology of optimal experience, volume 1990, Harper & Row New York, 1990.
[23] P. Haslum, N. Lipovetzky, D. Magazzeni, C. Muise, An introduction to the planning domain definition language, Synthesis Lectures on Artificial Intelligence and Machine Learning 13 (2019) 1–187.
[24] A. Vemuri, M. Heintzelman, A. Waad, M. L. Mauriello, K. Decker, G. Dominick, Towards dynamic action planning with user preferences in automated health coaching, Smart Health 28 (2023) 100389.
[25] P. Pirolli, S. Mohan, R. Yang, A. Venkatakrishnan, M. Silva, M. Youngblood, A. Ram, L. Nelson, User modeling and planning for improving self-efficacy and goal adherence in mhealth, Front. Public Heal. (2016).
[26] M. Gourlan, P. Bernard, C. Bortolon, A. J. Romain, O. Lareyre, M. Carayol, G. Ninot, J. Boiché, Efficacy of theory-based interventions to promote physical activity. a meta-analysis of randomised controlled trials, Health psychology review 10 (2016) 50–66.
[27] E. L. Deci, R. M. Ryan, Self-determination theory, Handbook of theories of social psychology 1 (2012) 416–436.
[28] E. A. Locke, Toward a theory of task motivation and incentives, Organizational behavior and human performance 3 (1968) 157–189.
[29] J. Chen, Flow in games (and everything else), Communications of the ACM 50 (2007) 31–34.
[30] J. Schell, The Art of Game Design: A book of lenses, CRC press, 2008.
[31] S. Rogers, Level Up! The guide to great video game design, John Wiley & Sons, 2014.
[32] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, D. Wilkins, PDDL—the planning domain definition language, Technical Report CVC TR-98-003/DCS TR-1165, Yale

Center for Computational Vision and Control, 1998.
[33] E. Scala, P. Haslum, S. Thiébaux, M. Ramirez, Interval-based relaxation for general numeric planning, in: Proceedings of ECAI, 2016, pp. 655–663.
[34] E. De Pellegrin, R. P. A. Petrick, Planning domain simulation: An interactive system for plan visualisation, in: Proceedings of the International Conference on Automated Planning and Scheduling, 2024, pp. 133–141.
[35] P. Van Gorp, R. Nuijten, 8-year evaluation of gamebus: status quo in aiming for an open access platform to prototype and test digital health apps, Proceedings of the ACM on Human-Computer Interaction 7 (2023) 1–24.
[36] J. F. Allen, Towards a general theory of action and time, Artificial intelligence 23 (1984) 123–154.
[37] B. R. Barricelli, E. Casiraghi, J. Gliozzo, A. Petrini, S. Valtolina, Human digital twin for fitness management, Ieee Access 8 (2020) 26637–26664.
[38] E. L. Deci, R. M. Ryan, Intrinsic motivation and self-determination in human behavior, Springer Science & Business Media, 2013.