Towards Personalized Medication Planning

Lee-or Alon, Hana Weitman, Alexander Shleyfman, Gal A. Kaminka

Computer Science Department, Bar-Ilan University, Israel
alonlee1@biu.ac.il, weitman@cs.biu.ac.il, alexash@biu.ac.il, galk@cs.biu.ac.il

Abstract

Personalized medication plans determine the selection, dosage, and administration schedule of medications, to achieve medical goals that are *specific to the patient and to its individual health constraints*. This paper introduces medication planning as a novel domain for planning, using PDDL+. We present alternative approaches to representing medication planning problems, and discuss experiments that raise opportunities and challenges for the planning community.

Introduction

Personalized medication planning is the process of generating a plan of drug administrations that meets a given set of medical goals that are specific to the individual patient. The planing process must take into account general health safety constraints, helpful or harmful interactions between drugs, and individual physiological differences in responses to medications. The resulting personalized medication plan defines *what drugs* are administered, *when*, and at *what dosage*: too little is ineffective; too much is toxic.

Medication planning is a complex process, manually carried out by healthcare professionals. Its complexity is often encountered in mitigating harmful drug interactions in patients with multiple diseases (Dawes 2010), or in *combination therapy*, where multiple medications are used to synergistically improves therapeutic effects, while minimizing side effects (Turan et al. 2019; Singh et al. 2020). Indeed, combination of drugs can result in effects no drug can achieve alone (von Maltzahn et al. 2011).

Alaboud and Coles (2019) introduced a restricted case of medication planning, where the goal is to maintain a level of a single medication in the body of a patient. Their work uses PDDL+ (Fox and Long 2006) to model the non-linear effects of the drug by assuming it follows an exponential decay curve, parameterized by the drug half-life (a common assumption in medicine). Recently, we described a more general case, whereby the planning process considers multiple drugs, arbitrary non-linear effects, and the interacting biochemical properties of drugs and the body; these are considered with respect to patient safety and the achievement of medical goals (Alon, Weitman, and Kaminka 2023). However, we did not report on a representation approach, nor on any planners capable of carrying out such planning.

Continuing the work initiated by Alon et al. (2023), we present a general comprehensive approach for general medication planning, using PDDL+ (Fox and Long 2006). In contrast to previous work, the approach accommodates the administration of multiple medications, even repeatedly. It allows representation of both medical achievement goals (medical goals) as well as sustaining desired effects over time (maintenance goals). In addition, it facilitates modeling of drug-body interactions and the consideration of multi-drug effects. It supports the differentiation of medications with respect to various biological sites, which we collectively refer to as *bio-sites*. Consequently, goals can be articulated in terms of whole-body effects or tailored to target particular target area, using *targeted medications*.

We investigate three distinct alternative PDDL+ representations of the general medication planning problem. These vary in their approaches to representing constraints and the computation of drug densities within different organs. Through experiments conducted on synthetic problem sets inspired by real-world medication plans, we examine the opportunities and challenges presented by these alternative representation approaches. Additionally, we evaluate the interaction between these representations and various search algorithms and heuristics utilized in numerical planning, analyzing their effectiveness and potential limitations.

Personalized Medication Planning

Broadly, our work falls within a general trend of using various AI tools to assist and personalize medical care. We focus on *medication planning*, which is concerned with selecting drugs to be administered, as well as determining the dosage and scheduling of the chosen medications. Below, we discuss medication planning in detail; the broader context and background literature is discussed in a related work section towards the end of the paper.

Medication planning involves medical goals that are specified in terms of properties of different organs (or the body taken as a whole), and it takes into account temporal pharmaceutical dynamics and kinematics. These combine information about rate of accumulation and clearance of drugs in different bio-sites with information about toxicity and per-

sonal health constraints and activities to meet target levels of the drug or its biological effects. The process *selects*, *determines dosage*, and *schedules* the administration of medications to a patient. It uses *pharmacokinetic models* (Gerlowski and Jain 1983) that predict how a drug is distributed through different bio-sites, and how this *biodistribution* changes over time (Fig. 1). The biodistributions are either established empirically, or using half-life parameters assuming exponential decay trajectories. The planning process also uses *pharmacodynamic models* (Felmlee, Morris, and Mager 2012), which detail the changes in bio-sites, and may also predict drug-drug interactions. Both models are also used to determine dosage and to plan the timing of administration of various medicines, sometimes repeatedly, to achieve medical goals.

A good example of medication planning in mice is presented in the study of von Maltzahn et al. (2011). The plan has two steps, each using a different drug. The first step begins by administering a targeted drug that attaches to tumors and induces coagulation. After 72 hours the drug naturally clears from the body, leaving the tumor bio-sites with elevated coagulation. Then, a second drug is administered, which has the tendency to accumulate in bio-sites with elevated levels of coagulation. After 24 hours, treatment by near-infrared irradiation to increase drug accumulation. This approach utilizes two different drugs, which can provide targeted treatment only in combination. The first medicine type can target tumors, but cannot be used for imaging or for heating. The second medicine type cannot target tumors.

Alaboud and Coles (2019) and Alaboud (2022) introduced an early version of medication planning, a special case allowing for *a single medication*, whose target levels in the patient's body must be maintained in face of patient daily activities (influencing the desired drug levels). Using PDDL+ (Fox and Long 2006), they model single-medication problems, where the goal is to maintain pain-relief effectiveness constant, in face of patient activities. The levels are tracked in the body, as a whole. Conversely, in this paper we discuss problems with arbitrary number of organs. The representations we suggest also allow for arbitrary number of medicines, each may be administered repeatedly if needed. The interactions of the drugs are modeled, so that the planner can avoid harmful interactions or exploit synergies to generate combination therapies. We allow for effects that differ between various organs. We discuss three different PDDL+ representation approaches and contrast their effects when used with different search algorithms and heuristics.

The Medication Planning Problem

We show how the general medication planning may be mapped to planning. We then discuss alternative PDDL+. formulations of the planning problem.

Medication Planning as Planning

We remind the reader of the common definition of a PDDL+ planning problem (Fox and Long 2006): A tuple $\langle \mathcal{V}, \mathcal{S}, s_0, \mathcal{C}, \mathcal{G}, \mathcal{A}, \hat{\mathcal{E}}, \hat{\mathcal{P}} \rangle$ where $\mathcal{V}$ is a set of state variables either propositional or numeric (collectively called fluents),

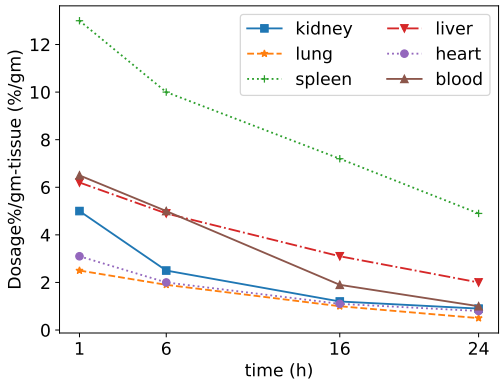


Figure 1: The biodistribution of a drug in mice, over time (derived from Akhtar et al. [2019]). The medicine levels as percentage of initial dosage are measured in the bloodstream, heart, liver, spleen, lung, and kidney, hours from the time of administration.

$\mathcal{S}$ is a set of states, where each state is a complete assignment of values to all variables $v \in \mathcal{V}$, $s_0 \in \mathcal{S}$ is an initial state, $\mathcal{C}$ is a set of constraints on possible assignments of values, and $\mathcal{G}$ is a goal description (a set of conditions over variables). $\mathcal{A}$ is a set of instantaneous actions that change the values of variables when selected by the agent, and $\hat{\mathcal{E}}, \hat{\mathcal{P}}$ sets of events and processes (resp.) that change the values of variables instantaneously or overtime, outside of the control of the agent. To the best of our knowledge, biologists' smallest measurement scale is minutes. Thus, we allow finite discretization of time.

Each action $a \in \mathcal{A}$ has a set of preconditions pre_a and a set of effects eff_a. If a state $s \in \mathcal{S}$ satisfies pre_a, action a is applicable and the agent may choose to execute it. When an applicable action is executed in s, the action changes state s variables according to eff_a.

A valid solution (a plan) is an ordered set of consecutively applicable *actions* that starts at the initial state s_0, transforms fluent values with each ordered action, and reaches a goal state, i.e., a state compatible with $\mathcal{G}$, such that no state in the state trajectory violates any of the constraints in $\mathcal{C}$. Typically, an optimal plan minimizes an objective such as number of actions, or their accumulating costs.

Planning for a Single Medication. We begin by mapping the medical components of medication planning to planning, for a single medication.

Fluents $\mathcal{V}$, and States. From a medical perspective, a patient's body is a set B of bio-sites, i.e., biological sites of interest, such as organs and blood. In simple pharmacological models, the whole body is considered as a single bio-site represents , while in advanced models, multiple bio-sites are modeled. The representation we propose is sufficiently general to accommodate multiple bio-sites.

Each bio-site is a set of P biochemical properties, whose values (indicating concentration levels or other measures of interest) generally vary between bio-sites. Each property of a given bio-site, is represented as a numeric fluent, whose value at any given time is measured in relevant standard units

(e.g., nanograms per gram of tissue). Parameters such as the levels of glucose in the blood are also represented in this way.

We use the set of fluents describing all properties in all bio-sites as the basis for $\mathcal{V}$. Properties interact locally (see below), and so while formally each fluent is independent, it makes sense to partition $\mathcal{V}$ according to bio-sites, and their properties, where $b[j]$ represents the property $j \in P$ in bio-site $b \in B$. Set $\mathcal{V}$ includes all such $b[j]$. It also includes other auxiliary variables—discussed later—used in tracking process dynamics, keeping track of administrations, etc.

A state is a complete assignment of values to all fluents in $\mathcal{V}$. Table 1 shows an example, where $\mathit{Liver}[p_2] = 0.001$, and $\mathit{Kidney}[p_2] = 4.3$. The *initial state* s_0 of a patient's body may be represented by setting the values of properties, in each bio-site, to current (normal or abnormal) values. For properties measuring drug concentration or accumulation, initial values in all bio-sites are zero.

Properties P \ Organs B	Blood	Heart	Liver	Spleen	Lung	Kidney
p_1	6.5	3.1	6.2	13	2.5	5
p_2	3.2	43.02	0.001	3.99	32.3	4.3

Table 1: Illustration of the state a patient's body at a given time. Columns represent organs. Rows represent property values.

Administrating Drugs: Actions $\mathcal{A}$ and Processes $\hat{\mathcal{P}}$. When a drug is administered it affect properties in several ways. First, it directly affects the corresponding properties tracking the levels of the drug itself in different bio-sites. Second, changes to these can trigger biochemical reactions that change (reduce or elevate) other properties.

Direct Effects. We begin by describing the direct effects of administrating a drug. The effects of this action are highly non-linear, and are often modeled in medical literature by an exponential decay with a half-life parameter, or by pharamacokinetic models (e.g., the k-compartment model (Gerlowski and Jain 1983)), that describe the biodistribution trajectories of the administered medicine, tracking its accumulation and clearance in various bio-sites over time. Every administration has a multidimensional (multiple bio-sites, and over time) continuous non-linear effect. This is the baseline trajectory of the medicine type accumulation behavior (*baseline* in short).

In Figure 1, we see the baseline trajectory for a single property, in multiple bio-sites (*blood, liver, kidney, heart, lung, spleen*), for an injection at time $t_0 = 0$. Such trajectories change of course between medicines, but may also change between patients. The approach we take in the PDDL+ implementation uses explicitly-encoded trajectories in the domain description, which can differ between patients.

A PDDL+ *action template* $m(d, t)$ represents administering dosage d of a drug m at time t.[1] This representation allows for multiple administrations of the same medicine type at different times, but not simultaneously.

[1]For simplicity, we distinguish between an action $m()$ and a medicine type m by including parentheses in the action label.

The selection of an action $m()$ triggers a PDDL+ process computing its effects. For every medicine m, there is *at least* a single property m in every bio-site b, i.e., $b[m] \in \mathcal{V}$, which represent the level of medicine m in bio-site b. The process computes the grounded effects of an action $m(d, t_0)$ as a function of the initial dosage d and the time since injection $t - t_0$, which yields a percentage in the associate biodistribution trajectory. We follow common practices, and assume that the baseline trajectories are given in percentages of the initial dosage, for a standard mass unit.

Formally, the baseline effect of an action $m(d, t_0)$ determines the values of properties $b[m]$ for any time $t \geq t_0$ and every bio-site $b \in B$. The process thus sets for any time step t:

$$b[m] := f_m(t - t_0, b) \cdot d \tag{1}$$

where $f_m(t - t_0, k)$ is the value of the biodistribution trajectory of medicine type m for bio-site b at absolute time t, when the administration took place in time t_0. A single process may be used to simulate the direct effects in P properties, in each of B bio-sites, by having $B \cdot P$ effects.

Indirect effects (on other properties). The direct effect of a drug is its accumulation in a bio-site, tracked by the associated property. This triggers and causes changes in other biochemical properties in the same bio-site (some of these changes are the purpose for administering the drug). Given a model of how values of property p_1 affect a different property p_2 in the same bio-site, the PDDL+ process can compute both direct and indirect effects. For space reasons, we avoid details here, and ignore indirect effects in the experiments.

Effects modified by other properties. The drug properties are influenced not only by the baseline effects of an action, but also by the dynamically-changing values of other properties. The von Maltzahn (2011) example previously discussed serves to illustrate: The baseline behavior of the second medicine changes in response to coagulation presence, which is why the first drug is used initially to increase coagulation in tumors. For now, we assume that such dependencies are *local*: property values in one bio-site are influenced solely by other properties within the same bio-site.

In planning terms, this means that the effects of an action $m(d, t)$, as computed by the associated PDDL+ process yield different values for $b[m]$, depending on some other properties $D \subseteq P$. This can be achieved by using separate processes (one per set of conditions) or, for brevity, with a single process using PDDL+ *conditional effects*. We add the biodistribution conditions to the process effects' preconditions segment, where each conditional effect specifies how the values change when the conditions are met. To prevent conflicting effects by overlapping conditions, we explicit consider all possible condition combinations. Given r conditions, this means 2^r condition combinations. Thus for instance, if $b[m]$ is set differently depending on two conditions $b[p_1] < x$, and $b[p_2] > y$, we must specify how $b[m]$ is set for all four condition outcomes.

Personalized Goals $\mathcal{G}$ and Safety Constraints $\mathcal{C}$. Given the definitions of states and actions above, it seems a simple matter to define goal states in terms of target levels for

properties of interest, at a specific set of bio-sites (therapeutic sites). However, medically, we must also ensure that the levels of all properties are maintained at safe levels, *before* the target levels are reached, as well as *after*.

We use constraints (Scala et al. 2016b) to impose limits on the maximal and/or minimal values of a property at any moment. These limits can come from medical defaults, or they may be personalized for specific health conditions of a patient. For example, if a patient has diabetes, the glucose level must stay below a given threshold h at all times. Such a constraint on the property j of bio-site b can be expressed as $b[j] \bowtie h$, where $\bowtie \in \{>, \geq, <, \leq\}$.

Constraints can be placed on the interactions between drugs. For example, we may represent a constraint that if a property value i in a bio-site b is greater than a given threshold h_i, the value of property j in the same bio-site must be less than a threshold h_j, i.e., $b[i] > h_i \Rightarrow b[j] < h_j$.

The goal description $\mathcal{G}$ has two components in the PDDL+ representation of medication planning. The first involves specifying target levels for properties in the set of therapeutic sites. These target levels can be personalized and differ between patients. The second component ensures that constraints are maintained after these target levels are achieved.

Once the goal conditions are first satisfied at time t_g, safety constraints should be upheld not only in the interval $[0, t_g]$ but also in the extended interval $[t_g, \infty)$, bearing in mind that action effects have finite durations. Thus a second subgoal introduced using PDDL+ checks that all administered medication had been eliminated from the patient's body *after* the first component has been achieved.

Multiple Medications We now turn to extending the representation to allow for multiple medication types (drugs), and repetitive administrations. We begin by considering multiple medications, each given at most once as part of a medication plan. The use of multiple drugs does not preclude simultaneous effects on the same bio-site property, either directly or indirectly. This is relatively straightforward to represent.

When more than one drug affect the same property, the change in the property value due to the medicine administration cannot be simply overwritten (as is done for the baseline in Eq. 1). Instead, the property values are computed by adding the various relative effects, due to the different medications.

Consider an administration of medicine m with a biodistribution as shown in Figure 1. For simplicity, we do not consider the administration dosage and other properties affects on this example. If the planner calculates the effect of this administration of the level of medicine m in the spleen two hours after administration, the planner will adjust the level of medicine m in the spleen relative to the previous time step (one hour post-administration). This adjustment represents the difference between the current accumulation level of m in the spleen (13) and its accumulation level at the previous time step (12.4). That is:

$$\textit{spleen}[m] \mathrel{+}= 12.4 - 13$$

We expand the representation to permit up to N administrations of the same type of medicine. Different medicines may be administered simultaneously. However, the administration times for any two instances of the same type of medicine must differ. As before, medications may simultaneously affect the same bio-site property.

We denoted the N potential administrations of medication m (medication instances) by $m_1, \ldots, m_N$. Due to PDDL+ limitations, N must be finite and known in advance: the PDDL+ files must prepare these N potential instances in advance. Each administration of an instance initiates a new process. However, we must differentiate between processes triggered by distinct administrations of the same medicine. Consequently, different administrations of the same medicine are enumerated and their effects are modeled as different process instances.

The administration action receives an instance m_i and a dosage amount. Two conditions must be met to perform the action: instance m_i has not been administered yet, and the state has not administered another instance of the same medicine type in the current time step. To keep track of which medicine type was administered in a time step, the state holds a predicate for each medicine type. At the initial state, all these predicates are initialized to *False*. When an instance i of medicine m is administered, the corresponding predicate is marked as *True*. An event resets these predicates to enable the administration of instances of these medicine types in the next time step. For simplicity, we omit the representation of this event in the discussion.

In addition to the medicine type predicate, each administration instance has also a predicate, which indicates whether that specific instance was already administered. Once an action chooses to administer instance m_i, it sets the respective predicate of that administration instance to be *True*. When *True*, the predicate activates a new process instant for the administration i of medicine type m. To model N administration instances of each medicine type, we need to define N variables for each medicine type, i.e., a total of $M \cdot N$ variables and instances of such predicates.

An Example Process with Repetitive Administrations We consider an example with M medication types, which affect a single bio-site property $b[p]$, under two conditions c_1 and c_2. We allow up to N repetitive administration of each medication types.

As discussed above, every administration of a drug $m \in M$ triggers a separate process. Below, we show the representation of *one* of these processes, using conditional effects that compute the change in property $b[p]$ due to the administration of medicine m_i, i.e., the i's administration of the same medicine type $m \in M$. To ease the presentation, we abuse the notation and use $d(m_i)$ to denote the dosage of the specific administration, and $t(m_i)$ to denote the time that has passed from the administration of medicine m_i. It is initialized to 0 on the administration action, and the process increments it by the time step size $\#t$. Each process (even of the same medicine type) has separate variables, except for the state property value $b[p]$, which is shared between all processes.

The function $g(b[p], m, c, t)$, describes the effect of medicine type m on the property $b[p]$ under condition com-

bination c at time t from the administration. It is a discretization of medicine m's biodistribution. The function itself is identical to all processes of medicine type m. However, the sampling time t is different for different repetitive administration of m. We assume g returns normalized values, which are then multiplied by the dosage $d(m_i)$.

As multiple medications may affect the same bio-site property simultaneously, the change in the property value due to the medicine administration cannot be simply overwritten, using the PDDL+ *assignment* keyword. Instead, we increment the change in the bio-site property due to the various administrations.

Specifically, in the conditional effects, the process calculates the difference between the current property level (that is, $d(m_i) \cdot g(b[p], m, c, t(m_i))$) and the previous property level created by this medicine instance, i.e., prev$(b[p], m_i)$, and adds it to the new property value. This way, the process considers **only the delta** added (or removed) from the previous effect by this administration instance, without overwriting other medicine effects. That allows multiple medicines to affect the same bio-site property simultaneously.

Figure 2 shows PDDL+-like pseudo-code of the example process described above, for the medication m_i, the i^{th} administration of medication m. For readability, we avoid the Lisp-like syntax. We emphasize again that such processes would be triggered for every administered instance i of every medication $m \in M$.

effect:
$t(m_i)$ += #t
when c_1:
$b[p]$ += $d(m_i) \cdot g(b[p], m, c_1, t(m_i))$ − prev$(b[p], m_i)$
prev$(b[p], m_i) := d(m_i) \cdot g(b[p], m, c_1, t(m_i))$
when c_2:
$b[p]$ += $d(m_i) \cdot g(b[p], m, c_2, t(m_i))$ − prev$(b[p], m_i)$
prev$(b[p], m_i) := d(m_i) \cdot g(b[p], m, c_2, t(m_i))$

Figure 2: PDDL+-like effect representation of a single process computing the conditional effects of m_i.

Action- and Event-based Representations

Aside from the major representation building blocks previously described for, there are other representation decisions to be made. These include the representation of safety constraints and medical goal achievement (i.e., achievement of target levels). Safety constraints can be directly represented by PDDL+ constraints (as the default, described above), but could also be replaced with events (see below). Book-keeping actions (described below) can also be replaced with events. This gives rise to the three following representations: actions-and-constraints (A&C), events-and-constraints (E&C), and events-and-events (E&E).[2]

[2]We omit the actions-and-events (A&E) representation due to space constraints. As seen in the following section, A&C performance is the worst among the given representations due to a higher branching factor. Since A&E shares actions with A&C, it performs similarly and does not provide us with important information.

E&C. When medicine m is administered, a corresponding process is triggered to simulate the medicine's effect on the patient's body. This process can either increase or decrease the current value of a property in the affected bio-sites. We monitor the highest attained value of the goal property in the therapeutic sites through events triggered when a new maximum is reached. These events record the updated maximum until it achieves the goal value.

Given that the biodistribution trajectory may not always be monotonically decreasing, a valid plan must wait until all medicines are cleared from the patient's body. Consequently, we also model the clearance occurrence as an event. When a medicine administration is cleared, the corresponding event is triggered, satisfying the relevant goal condition. This is the natural representation and we see two other representations as augmentations of this one.

A&C Despite the advantages of PDDL+ events, an event-based representation has limitations. Event conditions are evaluated after each time step, hindering accurate estimations by some heuristics. To overcome this, we addressed the issue by modeling higher-value events and cleared-out events as actions. This approach allows a search algorithm to update the current maximum seen value selectively when it brings the property closer to the goal, aligning with the objective of attaining specified values in each bio-site property without constraint violations. The same rationale applies to the cleared-out event, optimizing updates when both possible and beneficial. In the action-based representation, safety constraints are modeled as PDDL+ constraints, akin to their representation in the event-based E&C approach.

E&E This representation is similar to E&C, with the distinction that medical safety constraints are modeled as PDDL+ events. These events are triggered when a violation of medical constraints occurs. The activated event designates the state as a dead-end, halting all processes and rendering all actions inapplicable in this state.

Experiments

To evaluate the use of PDDL+ for medication planning, we empirically evaluate synthetic medication problems using the ENHSP-20 planner (Scala et al. 2020b), supporting PDDL+ with the exception of conditional effects. To circumvent this limitation, we converted each process conditional effect to multiple processes, and set the time step to one hour.

We used the following ENHSP-implemented numeric heuristics: *blind* (1 for a non-goal states, 0 for goals), the admissible heuristics h^{max} and h^{rmax}, and inadmissible h^{add} and h^{radd} are all due to Scala et al. (2020a). Their treatment of processes and events is based on the inadmissible *AIBR* heuristics that is due to Scala et al. (2016a). The search algorithms used are the A^* search (Hart, Nilsson, and Raphael 1968) and the Greedy Best First Search (*GBFS*) (Doran, Michie, and Kendall 1966). *Planners*, therefore, are a combination of a search algorithm and a heuristic, where opt-blind is $A^* + blind$, opt-hmax is $A^* + h^{max}$, opt-hrmax is $A^* + h^{rmax}$, sat-aibr is $A^* + AIBR$, sat-hadd is $GBFS + h^{add}$, and sat-hradd is $GBFS + h^{radd}$.

As a start, we examined the representation's ability to model personalized medical goals. We designed two problem instances with one medicine type with clearance time of ten hours, and four repetitive administrations. Both problems have one property bio-site of interest. The difference between the problems is the medical goal. In the first problem instance, the goal was to achieve a value greater than 100 in the bio-site's property, while the goal of the second problem was to achieve a value greater than 300 in the same bio-site property. We solved these problem instances using the opt-hmax planner. The planner suggested different plans. For the first problem, the planner suggested administering only two out of the four administrations, while for the second problem the planner suggested administering all the administrations.

We then turned to evaluating the three representation proposed in computational terms. We used 78 problem instances, differing in the number of medicine types M (1–3), potential repetitive administrations of each type N (1–4), the number of bio-sites B (1 or 2) and properties of interest P (1 or 2). We also vary the medication clearance time from the body, i.e., the duration of the biodiversity trajectory curves (7,10,13,17,20, or 24 hours); longer clearance times require the planner to maintain its associated processes over more states. The desired values in the goal may differ between problem instances. The experiments where performed on a machine with Intel Core i7-8700K 3.70GHz CPU and 32GB RAM.

We tested every problem with each of the different planners (algorithm+heuristic combination). Runtime was measured in CPU-seconds, using the Linux time command. If no solution was found after 30 minutes, the planning process was stopped; some planning processes ran out of memory. We limited the memory consumption to 16GB RAM. In both scenarios, the inability of the planner to find a solution was considered a failure. All tests were conducted using a single thread, meaning the experiments were performed one problem at a time.

The results are shown in Table 2. It displays the coverage (C), geometric means of runtimes in seconds (T), and the geometric means of the expanded nodes (N) for each representation. Coverage indicates the number of instances successfully solved by a planner out of 78 problems. The geometric means are computed on the runtime (and the expanded nodes) of the 37 problem instances solved by opt-blind in the A&C representation. This combination serves as the baseline, as every problem solved by opt-blind (A&C) was solved by all other planners, using every representation. That is, columns *C* and *T* present the result of the problem instances which **all** planners successfully solved (in this case, the results of the 37 problems opt-blind solved).

As one can see, the A&C representation has the lowest coverage and the largest runtime geometric mean, i.e., by using the A&C representation, planners solved fewer problem instances, and when succeeded in solving the problem, it took longer. The A&C representation should be more goal-oriented in the sense that an update does not require regularly checking events' conditions, but an update occurs only when possible and beneficial. However, its branching factor is higher in comparison to the other representations, thus the search process complexity is greater.

	A&C			E&C			E&E		
Planner	**C**	**T**	**N**	**C**	**T**	**N**	**C**	**T**	**N**
opt-blind	37	8.26	181024.79	48	**2.39**	**14990.58**	**49**	2.84	21243.23
opt-hmax	51	2.82	1236.05	62	**1.81**	**79.39**	**64**	1.85	82.57
opt-hrmax	43	3.46	1236.05	**59**	**1.86**	**79.39**	**59**	1.95	82.57
sat-aibr	48	3.14	905.03	54	**2.41**	**312.51**	**56**	2.6	371.37
sat-hadd	52	3.03	505.21	60	**1.93**	**128.71**	**65**	1.99	132.03
sat-hradd	53	3.22	479.40	59	2.07	**128.91**	**62**	**2.06**	131.58

Table 2: The coverage (C), geometric means of runtimes in seconds (T), and the geometric means of the expanded nodes (N) of the planners (in rows), across the three representations we suggest: actions-and-constraints representation (A&C), events-and-constraints (E&C), and events-and-events (E&E).

For most planners, the E&E representation has a higher coverage value than the E&C representation. The E&C representation models medical safety constraints as PDDL+ constraints, while the E&E representation models them as events. The E&C representation is smaller in comparison to the E&E representation. Therefore, the planner is able to run faster using the E&C representation for the smaller problems (thus, performs a lower runtime geometric means). Nevertheless, the tested planners do not consider PDDL+ constraints in their heuristic function calculations. Hence, PDDL+ constraints do not guide the search towards goal states. Modeling these constraints as events, allow the heuristic functions to exploit the event effects on the state to direct the search. Therefore, the E&E representation solved more problems than the E&C representation, while E&C has the lowest geometric means of expanded nodes for all planners among the problems that were solved by all planners.

Planners opt-hmax and sat-hadd have the highest coverage values among the optimal planners and the satisfying planners, respectively, across all representations (besides *sat-hradd*, which performs slightly better under the A&C representation). As shown in Table 2, opt-hmax and opt-hrmax expand the same number of nodes, and sat-hadd and sat-hradd are extremely close in terms of expanded nodes for all representations. However, opt-hmax and sat-hadd exhibit significantly better coverage than their 'r' counterparts. We believe this happens due to the fact that opt-hrmax and sat-hradd utilize redundant constraints (Scala et al. 2020a). Since the Medication Planning Problem domain does not employ redundant constraints, we observe that opt-hmax and sat-hadd are strictly better in terms of speed of calculation than their variations that account for redundant constraints.

Focusing on the leading planners, Figure 3 contrasts the runtime of opt-hmax and sat-hadd on all problems. The two axes measure the runtime for each problem *on log-scale*. Each marker represents a single problem. A marker below the main diagonal indicates that sat-hadd run faster than opt-hmax on the specific problem. A marker above the main diagonal indicates that opt-hmax had a lower runtime. When a run was unsuccessful for a planner, it is given a runtime of 1900 seconds (denoted by the red lines). Multiple mark-

ers can be hidden by another marker if they share the same runtime (for example, by the marker at the top right corner at (1900,1900), which indicates all the unsuccessful runs by both planners).

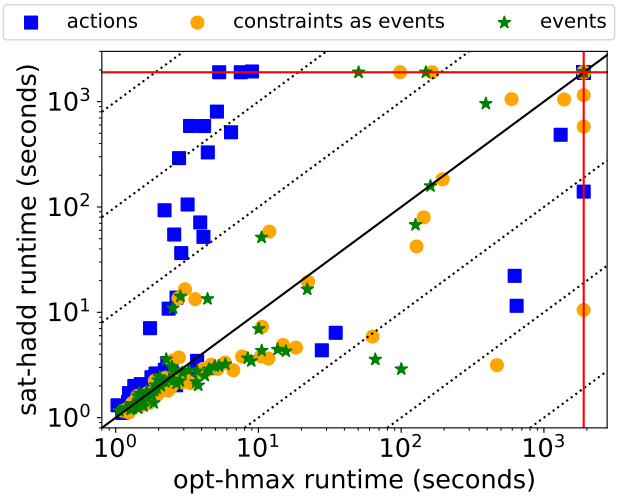


Figure 3: The runtimes of opt-hmax compared to sat-hadd runtimes, measured on log-scale axes. Each marker represents a single experiment. Unsuccessful runs are marked as a runtime of 1900 seconds for spacing. All problems where both opt-hmax and sat-hadd failed are located at the upper right corner of the graph.

Generally, opt-hmax performed better with the A&C representation in comparison to sat-hadd. Additionally, in most experiments, sat-hadd yielded better runtimes using the E&E representation. Since the E&E representation performed the best in means of coverage, for the rest of this section, we focus on this representation.

Figure 4 presents the opt-hmax runtimes compared to sat-hadd runtimes using the E&E representation, in a similar fashion to Figure 3. Here, the markers distinguish between the number of medicine types used in the problem (i.e., the size of M). As before, unsuccessful runs are marked as a runtime of 1900 seconds (denoted by the red lines).

Generally, adding medicine types increases these heuristic runtimes by orders of magnitude. It appears that sat-hadd handles two and three medicine types better than opt-hmax, while opt-hmax can deal better with a single medicine type.

While Figure 4 shows an exponential growth in the number of medicine types, it tells only a part of the story. The problem sizes and the runtimes do not depend on the number of medicine types alone, but also on the number of repetitive administrations N, and the medicine clearance times.

Figure 5 presents the opt-hmax and sat-hadd runtimes, for different numbers of allowed repetitive administrations (N), as a function of medicine clearance times. Both parameters generally affect the runtime of the planners. However, the allowed number of repetitive administrations has a greater effect over the runtime, as we see that any increase in N translates to clearly higher run-times. This parameter increases the branching factor, thus increasing the search space.

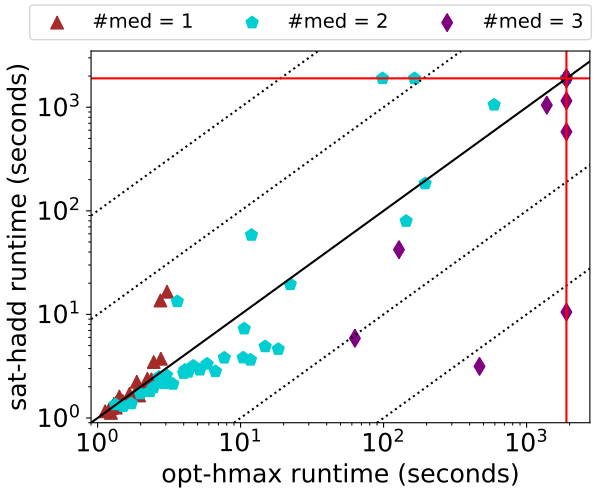


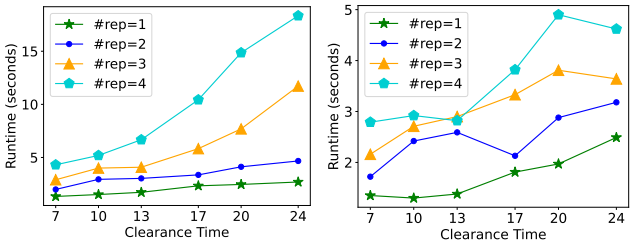
Figure 4: Runtimes of opt-hmax and sat-hadd in the E&E representation, on log-log axes. Each marker represents a single experiment. Problems are categorized by the number of medicine types. Unsuccessful runs are placed on 1900 seconds for spacing. These limits are marked by the red lines. All problems where both hmax and hadd failed are located at the upper right corner of the graph.

Related Work

Personalized medication planning is a recent area of research within the more general area of AI use personalized treatment planning, tailoring therapeutic interventions to individual patient and medical needs.

A prime example of treatment planning is *radiation therapy planning*, whereby motion planning and machine learning methods aid in the design of radiation treatments by simulating precise configurations, such as visualizing affected tissues, and suggesting treatment parameters that align with medical objectives. This contributes significantly to the continuum of patient care, as highlighted in recent reviews by Wang et al. (2019), Chow (2022), and Jones et al. (2023). Due to significant variations in tumor morphology, position, and other patient-specific factors, personalized radiotherapy plans must be formulated in order to minimize damage to normal tissue while persevering sufficient tumor control.

Another example of this general treatment plan is in planning treatments for patients with multiple diseases, by merging available multiple single-disease clinical guidelines. This intricate process includes substituting drugs when adverse or redundant interactions occur, adjusting and scheduling tests to monitor for such interactions, and other related tasks. Techniques for (partially) automating this process utilize constraint satisfaction (Wilk et al. 2013; Piovesan and Terenziani 2016), model-based reasoning (Riaño and Collado 2013; Piovesan and Terenziani 2015), and planning (Sánchez-Garzón et al. 2013; Fdez-Olivares et al. 2019; Michalowski et al. 2021). While these investigations touch of medicine choices, they do not provide personalized dosage or hourly medication schedule. Instead, they produce plans that span weeks or months, rather than hours.



(a) opt-hmax. (b) sat-hadd.

Figure 5: Runtimes of (a) opt-hmax and (b) sat-hadd using the E&E representation of problems with two medicine types, and one bio-site with two properties of interest. The vertical axis shows the medicine clearance times. The horizontal axis shows the runtime in seconds.

Similarly, the use of HTN planning to schedule chemotherapy treatments over weeks, considering availability of doctors (Gonzáez-Ferrer et al. 2013), is different from general medication planning as discussed here.

A final example area of treatment planning is discussed by Amir et al. (2015) and Amir, Grosz, and Gajos (2016). They use teamwork theory to enhance and improve collaboration between patients and various caregivers: family, medical professionals, and support organizations; the improvements allow the caregivers to better coordinate their human decision-making.

Conclusion

We propose the use of PDDL+ to represent medication planning problems, a relatively new and underexplored area in medical treatment planning. This domain requires planning for actions whose effects are durative, multi-dimensional and non-linear, which may overlap in time. In this study, we examined three variants of task representation: actions-and-constraints (A&C), events-and-constraints (E&C), and events-and-events (E&E). Each representation models medical safety constraints and goal-reaching decisions differently. The experiments show that the E&E representation performed best in terms of coverage. The best results among planners were obtained by opt-hmax and sat-hadd. While opt-hmax provides an optimal solution and sat-hadd guarantees only problem satisfaction. The fact that two planners had similar geometric means of runtimes leads us to believe that in some problem instances, we can obtain an optimal solution almost as quickly as any solution.

Medication planning presents a promising new avenue for planning, posing significant challenges: ideally, it would involve dozens or hundreds of drugs, some of which accumulate effects over days and weeks with frequent administrations (many psychiatric drugs exhibit this property). Furthermore, efficient representation of drug-drug and drug-body interactions that are currently modeled by conditional effects is necessary in this domain.

Much work remains for the future. For instance, exploring the suggested representation using various planners, such as Metric-FF (Hoffmann 2003) and OPTIC (Benton, Coles, and Coles 2012), requires translating the suggested representation from PDDL+ to PDDL. This translation can be facilitated by dedicated translators, such as the one proposed by Percassi et al. (2023). The process effects in the PDDL+ format should ideally be expressed as a multiplication of some expression and $\#t$, as outlined by Fox and Long (2006). However, in our suggested representation, although it aligns with the input standards of ENHSP, the process samplings of biodistributions have effects that deviate from the format specified by Fox and Long. Addressing this discrepancy and harmonizing the process modeling with the established framework presents an avenue for future investigation.

Acknowledgments This research was supported in part by ISF grant #2443/23 (Shleyfman). As always, G.K. thanks K. Ushi.

References

Alaboud, F. K. 2022. *Personalising Medication & Activity Regimes Using Novel State Progression Models for Forward Search with PDDL+*. Ph.D. thesis, King's College London.

Alaboud, F. K.; and Coles, A. 2019. Personalized medication and activity planning in PDDL+. In *ICAPS*, 492–500.

Alon, L.; Weitman, H.; and Kaminka, G. A. 2023. First Steps Towards Planning for Targeted Medicine. In *KEPS*.

Amir, O.; Grosz, B.; and Gajos, K. 2016. Mutual Influence Potential Networks: Enabling Information Sharing in Loosely-Coupled Extended-Duration Teamwork. In *IJCAI*, 796–803.

Amir, O.; Grosz, B.; Gajos, K.; Swenson, S.; and Sanders, L. 2015. From Care Plans to Care Coordination: Opportunities For Computer Support of Teamwork in Complex Healthcare. In *CHI*.

Benton, J.; Coles, A.; and Coles, A. 2012. Temporal planning with preferences and time-dependent continuous costs. In *ICAPS*, volume 22, 2–10.

Chow, J. 2022. *AIME*, chapter Artificial Intelligence in Radiotherapy and Patient Care, 1275–1286. Springer.

Dawes, M. 2010. Co-morbidity: we need a guideline for each patient not a guideline for each disease. *Family Practice*, 27(1).

Doran, J. E.; Michie, D.; and Kendall, D. G. 1966. Experiments with the Graph Traverser program. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 294(1437): 235–259.

Fdez-Olivares, J.; Onaindia, E.; Castillo, L.; Jordán, J.; and Cózar, J. 2019. Personalized conciliation of clinical guidelines for comorbid patients through multi-agent planning. *AIME*, 96: 167–186.

Felmlee, M. A.; Morris, M. E.; and Mager, D. E. 2012. Mechanism-based pharmacodynamic modeling. *Computational toxicology: volume I*, 583–600.

Fox, M.; and Long, D. 2006. Modelling mixed discrete-continuous domains for planning. *JAIR*, 27: 235–297.

Gerlowski, L. E.; and Jain, R. K. 1983. Physiologically-based pharmacokinetic modeling: principles and applications. *Journal of pharmaceutical sciences*, 72(10): 1103–1127.

Gonzáez-Ferrer, A.; ten Teije, A.; Fdez-Olivares, J.; and Milian, K. 2013. Automated generation of patient-tailoed electronic care pathways by translating computer-interpretable guideslines into hierarchical task networks. *AI in Medicine*, 57: 91–109.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.*, 4(2): 100–107.

Hoffmann, J. 2003. The Metric-FF Planning System: Translating "Ignoring Delete Lists"to Numeric State Variables. *JAIR*, 20: 291–341.

Jones, S.; Thompson, K.; Porter, B.; Shepherd, M.; Sapkaroski, D.; Grimshaw, A.; and Hargrave, C. 2023. Automation and artificial intelligence in radiation therapy treatment planning. *JMRS*.

Michalowski, M.; Rao, M.; Wilk, S.; Michalowski, W.; and Carrier, M. 2021. MitPlan 2.0: Enhanced Support for Multimorbid Patient Management Using Planning. In *AIME*, 276–286.

Percassi, F.; Scala, E.; and Vallati, M. 2023. A practical approach to discretised PDDL+ problems by translation to numeric planning. *Journal of Artificial Intelligence Research*, 76: 115–162.

Piovesan, L.; and Terenziani, P. 2015. A Mixed-Initiative Approach to the Conciliation of Clinical Guidelines for Comorbid Patients. In *AIME: Knowledge Representation for Health Care*, volume 9485 of *LNCS*, 95–108.

Piovesan, L.; and Terenziani, P. 2016. A Constraint-Based Approach for the Conciliation of Clinical Guidelines. In *IBERAMIA*, volume 10022 of *LNCS*, 77–88.

Riaño, D.; and Collado, A. 2013. Model-Based Combination of Treatments for the Management of Chronic Comorbid Patients. In *AIME*, volume 7885 of *LNCS*, 11–16.

Sánchez-Garzón, I.; Fdez-Olivares, J.; Onaindía, E.; Milla, G.; Jordán, J.; and Castejón, P. 2013. A Multi-agent Planning Approach for the Generation of Personalized Treatment Plans of Comorbid Patients. In *AIME*, 23–27.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramirez, M. 2016a. Interval-based relaxation for general numeric planning. In *ECAI*, 655–663.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramírez, M. 2020a. Subgoaling Techniques for Satisficing and Optimal Numeric Planning. *JAIR*, 68: 691–752.

Scala, E.; Ramirez, M.; Haslum, P.; and Thiébaux, S. 2016b. Numeric planning with disjunctive global constraints via SMT. In *ICAPS*, 276–284.

Scala, E.; Saetti, A.; Serina, I.; and Gerevini, A. E. 2020b. Search-Guidance Mechanisms for Numeric Planning Through Subgoaling Relaxation. In *ICAPS*, 226–234.

Singh, S. K.; Mohammed, A.; Alghamdi, O. A.; and Husain, S. M. 2020. New approaches for targeting drug resistance through drug combination. In *Combination Therapy Against Multidrug Resistance*, 221–246. Academic Press.

Turan, O.; Bielecki, P.; Tong, K.; Covarrubias, G.; Moon, T.; Rahmy, A.; Cooley, S.; Park, Y.; Peiris, P. M.; Ghaghada, K. B.; et al. 2019. Effect of dose and selection of two different ligands on the deposition and antitumor efficacy of targeted nanoparticles in brain tumors. *Molecular pharmaceutics*, 16(10): 4352–4360.

von Maltzahn, G.; Park, J.-H.; Lin, K. Y.; Singh, N.; Schwöppe, C.; Mesters, R.; Berdel, W. E.; Ruoslahti, E.; Sailor, M. J.; and Bhatia, S. N. 2011. Nanoparticles that communicate in vivo to amplify tumour targeting. *Nature materials*, 10(7): 545–552.

Wang, C.; Zhu, X.; Hong, J. C.; and Zheng, D. 2019. Artificial Intelligence in Radiotherapy Treatment Planning: Present and Future. *Technology in Cancer Research & Treatment*, 18: 1–11.

Wilk, S.; Michalowski, W.; Michalowski, M.; Farion, K.; Hing, M. M.; and Mohapatra, S. 2013. Mitigation of adverse interactions in pairs of clinical practice guidelines using constraint logic programming. *JBI*, 46(2): 341–353.