Cancer Hallmark Automata (I): Model, Therapy and Complexity

Loes Olde Loohuis Andreas Witzel Bud Mishra

January 10, 2012

Abstract

This paper introduces Cancer Hallmark Automata, a formalism to model the progression of cancers through discrete phenotypes (so-called *hallmarks*). The classification of various cancers using stages and hallmarks has become common in the biology literature, but primarily as an organizing principle, and not as an executable formalism. The precise computational model developed here aims to exploit this untapped potential, namely, through automatic verification of progression models (e.g., consistency, causal connections, etc.), classification of unreachable or unstable states (e.g., "anti-hallmarks") and computer-generated (individualized or universal) therapy plans. The paper builds on a *phenomenological* approach, and as such does not need to model the biochemistry underlying the progression. Rather, it abstractly models transition timings between hallmarks as well as the effects of drugs and clinical tests, and thus allows formalization of temporal statements about the progression as well as notions of timed therapies. The model proposed here is ultimately based on *hybrid automata* (with multiple clocks), for which relevant verification and planning algorithms exist in the literature. By establishing a suitably expressive formalism, this paper also prepares the readers for its sequels, which plan to explore deeper algorithmic connections, model building techniques and realistic clinical applications.

1 Introduction

1.1 Background

Cancer is generally thought of as a *progressive disease* – in particular, a disease which has certain *discrete* states through which it progresses towards a full-blown final phenotype (e.g., metastasis). This view is reflected in the so-called *hallmarks of cancer* proposed by Hanahan and Weinberg [17], and it has become one of the predominant ways of thinking about cancer, solidified through many further publications and experiments. A recent article by the same authors [18] reviews and consolidates the new insights of the last decade.

According to that model, tumors must necessarily acquire certain "intermediate" hallmarks culminating in the "final" hallmarks of tissue invasion and metastasis. As the authors write,

> Simply depicted, certain mutant genotypes confer selective advantage on subclones of cells, enabling their outgrowth and eventual dominance in a local tissue environment. Accordingly, multistep tumor progression can be portrayed as a succession of clonal expansions, each of which is triggered by the chance acquisition of an enabling mutant genotype. [18, p. 658]

The current list of cancer hallmarks includes the abilities to reproduce autonomously, to ignore anti-growth signals, or to signal for formation of new blood vessels, as well as some other phenotypes. Hallmarks can be obtained in various different orders, but not every order is viable. Intuitively, a hallmark can be acquired by a certain population of cells if it conveys a selective advantage compared to the predominant phenotype in that population. For example, in a wildly growing cluster of cells, the ability to signal for new blood supply, and thus nutrients, oxygen, and waste disposal, will allow the respective sub-population to outgrow the others.

Most hallmarks are acquired through mutations of very specific sets of genes, and many of the targeted drugs that have been developed in recent years influence the function of the products of these genes [31]. For example, the vascular endothelial growth factor (VEGF) signals for creation of new blood vessels (*angiogenesis*), and the drug Avastin inhibits the associated signaling pathway, thus preventing growing tumors from obtaining the needed blood supply.

1.2 Motivation

The view of cancer progression and therapy bears a striking resemblance to formal models of state-transition machines in computer science. While cancer biologists obviously think in these concepts, they do not have, or aim at, such formal models. In this paper we present a formal framework called *Cancer Hallmark Automaton* (CHA) that allows us to formally capture cancer progression through accumulation of successive hallmarks. States of these automata represent hallmarks, and directed edges among pairs of states define paths, representing successive hallmark acquisitions.

Drugs can then be thought of as inhibiting specific transitions in a hallmark automaton. This simple model prompts a further application of computer science techniques, namely controller synthesis, to aid in the increasingly complex task of designing therapeutic plans for cancer treatment. A therapy can be designed to satisfy a broad range of goals. For example, one may require that the system stays within a specified region of the state space or satisfies a given temporal formula. The controller synthesis problem then consists in finding a timed therapy that manipulates the model in such a way that the desired behavior is satisfied.

In the following we describe our motivation to investigate cancer hallmark automata in more detail.

Useful abstraction level The hallmark view models carcinogenesis abstractly as a progression through distinct phenotypes, following particular traces. It is useful for two reasons. On the one hand, it is abstract enough to allow practically all forms of cancers to be analyzed in one comprehensive and intuitive framework – without getting bogged down by their complexity. On the other hand, it retains a sufficient level of detail to connect these phenotypes to specific genotypes and thus, to important low-level mechanisms involved in gene regulation, metabolism and signaling, some of which are accessible to various therapeutic agents.

Advantage of formalization Hallmark models are currently used both to establish diagnostic categories, and to inform therapeutic decisions. By making all assumptions explicit and establishing a formal framework, we hope to better understand the disease and its progression as well as its resilience against therapeutic interventions. For example, cocktail drugs are currently typically put together through trial and error. We expect our formal model to help identify more rigorously exactly which (parallel or back-up) paths in cancer progression need to be perturbed/blocked by the cocktail. We also wish to understand, by including time in our formal framework explicitly, how cancer progression can be slowed down to the point that it is manageable as a *chronic disease*, rather than cured completely. Such an approach may be preferable since it requires lower levels of toxicity.

Advantage of computation Even though current literature in cancer biology typically lists no more than a dozen hallmarks, the resulting models are not necessarily simple or easy to analyze, and future progression models may make more fine-grained distinctions between hallmarks (see next point). Also, the list of targeted drugs has grown enormously, and the task of finding a (near-)optimal therapy plan is soon to be beyond manual planning. An additional increase in complexity results from combinatorial notions like synthetic lethality [30] and cocktail drugs, or path-dependent notions like oncogene-addiction [43]. Automated computational tools will help tackle these complexities. Model checking can be used to automatically verify hypotheses about cancer progression, and controller synthesis to generate suggestions for therapeutic plans, possibly personalized based on patients' genetics.

Connecting to data Starting from a rigorous formulation of cancer hallmark progression models, we aim to take further steps towards implementing a practical system. The model we introduce is hoped to pave the way for both the automatic generation of fine-grained hallmark models from data (e.g., The Cancer Genome Atlas project[1]) and their systematic usage in cancer diagnostics and therapeutics.

[1] http://cancergenome.nih.gov

Model-discovery tools such as GOALIE [41] can be used to generate models from data using any desired resolution for the state space. Ultimately, these models are expected to be used for a wider variety of purposes, some discussed here, and some yet to be imagined: for example, mining clinical data to discover "bottlenecks" in cancer progression that point to promising drug targets, developing personalized models for specific cancers and patients, or even creating "expert systems" for clinicians and pathologists to query patient health records.

1.3 Approach

The current paper is the first one in a sequence and starts by motivating and defining cancer hallmark automata, suggesting further extensions and providing some preliminary algorithmic considerations. The framework proposed in this paper evolves gradually from a simple Kripke model by the successive addition of:

(i) "costs" of drugs in various dimensions (e.g., toxicity, side effects, eventual resistance, mode and frequency of delivery, discomfort, monetary cost, etc.) that are to be optimized,

(ii) timing of transitions and drug effectiveness,

(iii) partial observability of the tumor's internal state along with tests that can provide additional information about the state, as well as

(iv) the possibility of factoring in other parts of the tumor's host organism which may be affected by a therapy (e.g., stroma, liver, immune system, stem cells, etc.).

2 Related Work

In the following, we briefly review the relevant literature from the two fields that are combined: biology and computing.

2.1 Biology literature

The hallmark view of cancer, originally proposed by Hanahan and Weinberg in [17], and subsequently further modified [18], is the idea that carcinogenesis proceeds through a series of discrete phenotype states or hallmarks. We refer to the recent survey article [18] and the numerous references therein, for the developments of the last decade.

Most hallmarks are acquired through mutations of specific sets of genes, while global genomic instability drives the tumor progression through these hallmarks. These hallmark principles are also highly relevant to the development of targeted therapies. See Table 1 for a small sample of therapeutic agents that attack specific hallmarks based on the product of the genes that they influence [31].

Typically, these therapeutic agents are thought to act on the pathways that are behaving abnormally in the hallmark and thus restore normalcy (as in EGFR-mutated lung adenocarcinoma treated with Tarceva, or VEGF-mutated colorectal, lung, kidney, and glioblastoma cancers treated with Avastin). In these models, there is no notion of time or history of the cancer progression.

Combinatorial approaches to cancer therapy can also help to improve treatment of the disease [31]. For example, by combining several drugs affecting different mechanisms, or different signaling pathways used in a heterogeneous population, the progression to a next hallmark can be prolonged (or prevented). Our therapy-planning approach allows for searching over the combinatorial space of cocktail drugs.[2]

Finally, many other parts of the tumor's host organism influence, or are influenced by, the tumor's progression and therapeutic agents: stroma, liver [40], immune system [14], stem cells, etc. These parts of the host organism can be factored into our framework by modeling them separately and creating a suitable product automaton. It is then possible to model the effects of a therapy on the whole system, describe interactions between subsystems and specify therapeutic goals over all of them. To illustrate such a composite model, we formulate a liver automaton which progresses through different states of damage depending on the toxicity of the given drugs, and show how this can be combined with a CHA. The goal of a therapy is then to treat cancer without adversely affecting the liver. Other factors such as metabolic stress can also influence cancer progression [29, 31], but can be incorporated into CHA components *mutatis mutandis*, as will be described in a future paper in this series.

Agent	Target	Hallmarks	References
ABT-737	BCL-CL, BCL-2	EvAp	Stauffer, 2007
Alvocidib	CDKs	SSG	Lee and Sicinski, 2006
Bevacizumab	VEGF	Ang	Folkman, 2007
BEZ235	PI3K	SSG, Ang	Maira et al., 2008
GRN163L	hTERT	LRP	Dikmen et al., 2005; Harley, 2008
Nutlin-3	HDM2	EvAp, IAG	Vassilev, 2007

Table 1: Some therapeutic agents attacking specific hallmarks. Adapted from Luo et al. [31, Table 1], who give an extensive list and full references. We briefly explain the relevant hallmarks in Section 5.

[2]Note that we do not model cell types and heterogeneity explicitly, but its effects can be represented by having multiple outgoing transitions from a particular hallmark, each corresponding to the acquisition of a new hallmark by one of the sub-populations. See also Section 10.1.

2.2 Computer science literature

Automata represent formal frameworks to describe the (non-deterministic) behavior of discrete-state systems. These frameworks range from simple finite automata, where states are described by nodes and transitions by edges, to complex state machines involving real-time progression and partial observability. See, e.g., the book by Hopcroft et al. [25] for a general introduction.

Timed Automata extend classical automata to model progression of real-time systems. A timed automaton is a finite automaton with a set of real-valued variables, called clocks. Clock constraints on the edges and clock invariants at the states are used to restrict the possible progressions of the system. We refer to [2, 4] for an overview of timed automata.

Hybrid Automata further extend timed automata to allow for non-synchronous continuous evolution. More precisely, while in timed automata clocks increase synchronously at the same rate, clocks in so-called hybrid automata can run at different rates, which can change independently with the transition to another state. For an overview of the theory of hybrid automata see [19].

Stochastic Automata are stochastic state machines which satisfy the Markov property, i.e., their evolution only depends on the current state and not on the whole history of visited states. In that sense, they also belong to the paradigm of automata. Markov models exist in a variety of forms. They can allow for partial observability (HMMs, Hidden Markov Models [39]), for external control of the system's progression (MDPs, Markov Decision Processes [37]) or both (POMDPs, Partially Observable MDPs [33]).

System verification, and in particular model checking [12], is concerned with formally verifying whether a given system satisfies a given property. Such a property could pertain to the (non-)reachability of certain good or bad states, or, more generally, be any temporal statement about visited states. Typically, a temporal logic like Computation Tree Logic (CTL, [10]) is used to express properties. There exist many extensions of CTL, of which a particularly useful one for our purposes is Timed CTL [1]. It allows statements not only about the qualitative temporal order of visited states, but also includes quantitative temporal operators. Timed CTL can be generalized further to reason about costs and the values of different clocks [7]. System verification becomes more difficult when the behavior of the system is not fully observable. In such situations of partial observability, the observer can narrow down the possible states of the system to a subset of states, but the exact state may not be known. In many formalisms that include partial observability, it is assumed that the observer is "automatically" notified whenever the currently available observation changes.

Control theory and planning In control theory the objective is to manipulate a system ("plant") in such a way that the controlled system ("plant + controller") satisfies a certain desired specification. Much work has been done on automatically generating controllers for untimed automata (see [28] for an algorithm that uses CTL specifications) as well as timed automata (see, e.g., [3, 24]). More recently, timed control with partial observability has received more attention (e.g., [8]). Cassez et al. [9] show an efficient on-the-fly controller synthesis algorithm for timed automata with partial information. In hybrid automata theory, methods have been developed to design controllers for specific properties like non-reachability of bad states (so-called *safety properties*) as well as more general properties expressed in temporal logics, both using continuous- and discrete-time control [20, 22].

In the planning literature, algorithms have been developed to construct and validate action plans so that the resulting behaviors satisfy complex temporal formulas, called *(temporally) extended goals*. For example, in [6] planning under partial observability is studied, and in [38] planning in real-time systems.

Our contributions Our cancer hallmark automaton is shown to be a special case of a hybrid automaton. In the absence of reliable and sufficiently large amounts of data on expected outcomes from clinical tests, we have shifted our initial focus away from probabilistic models, and towards nondeterministic models, which capture uncertainty about state transitions, but lack quantitative probability distributions. Thus, the resulting therapies are highly conservative as they allow planning against the worst-case behavior, rather than the average or expected case. Still, our framework is justified by underlying stochastic processes involved in somatic evolution, see Section 4. Hybrid automata (rather than simple timed automata) are thus ideal for modelling how drugs may slow down those processes and thus affect their stopping times. Various drugs affect the transition to possible next states in different ways; we use multiple clocks to capture these processes.

Besides model-checking (verifying whether a certain set of properties is satisfied) these models of cancer progression, we are interested in their manipulation, so we proceed to develop notions of therapeutic intervention against the progression of cancer – taking into account its partial observability. We define notions of therapy, remaining consistent with various notions originally defined by control theorists in similar contexts. For example, a therapy is defined as a function from the set of runs to the set of controller actions, namely, administration of drug cocktails and medical tests. In Section 7 we show how this notion of therapy can be translated into a conditional plan as is common in the planning literature. We use CTL and its extensions to specify therapeutic goals as in some related works; and we model partial observability, but our convention for obtaining observations differs from the literature in that they are not automatically emitted by the plant but be actively obtained through test actions.[3] In Section 9 we describe the

[3]This point is mainly conceptual, since a system with test-based observations can be translated into one with automatic observations using additional states.

underlying controller synthesis algorithms, adapting those for hybrid automata developed in the control theory literature.

3 Overview

The rest of this paper is organized as follows. In the next section we start by outlining the basic assumptions underlying CHAs. Then, in Section 5, we introduce a basic CHA formalism. In this section, a CHA is modeled as a *finite non-deterministic automaton* whose nodes represent hallmarks and whose directed edges represent transitions from one hallmark to the next. The edges are labeled with drugs that can inhibit the transition. A *therapy* is defined as a function that assigns a set of drugs to each finite progression history, or *run*. An execution of a therapy is defined as a run of the CHA that respects the therapy, that is, no transition of the execution is inhibited by the therapy. Our model includes costs by associating a cost vector with each state and each cocktail. Therapies may be selected by comparing costs of possible executions using a notion of Pareto dominance, in addition to the required qualitative properties specified in CTL.

In Section 6 we extend the CHA framework to include real time. In this model, transitions take certain durations of time, and drugs can prolong (or stop) the transition process. This is modelled using a hybrid automaton with multiple clocks. Clock constraints on the edges and clock invariants at the states restrict the possible progressions of the system. Multiple clocks are needed to allow for the scenario that a drug affects the transition to possible next states in different ways. Possible runs and therapies of a timed CHA now include the clock values. An extension of CTL, Timed CTL, is used to specify extended goals about the system.

In Section 7 we introduce uncertainty into the framework. The oncologist may have only partial knowledge about the tumor's internal state, which we model by keeping track of his belief set. Tests are incorporated into the definition of a therapy as actions that reduce uncertainty about the current state. In our framework, tests have costs, but take no time. To integrate the observer's information about the system, we add epistemic operators to Timed CTL. In Section 7.4 we give a translation from therapies for timed CHAs with partial observability into conditional plans.

In Section 8, we present a simple liver automaton as an example of a system of the host organism that may be affected by the therapy. These systems can be combined with the CHA using parallel composition.

In Section 9, we discuss various algorithmic issues related to generating therapies, i.e., controller synthesis for CHAs.

Finally, Section 10 concludes with a discussion and an outline of the plans for the sequels of this paper.

4 Underlying Assumptions for Timed CHAs

There are a few tacit assumptions that we have used to motivate our model (especially, timed CHAs) and to a large degree, the structure of a therapy. Though all of of these assumptions are known to and largely accepted by the cancer biology community, for the sake of completeness, they are made explicit below.

1. Our models do not concern themselves with the *origin of a cancer*: e.g., we do not assume that cancer is a disease of the genome, initiated by a gain-of-function mutation in an oncogene or a loss-of-function mutation in a tumor suppressor gene, or a disease of aberrant signaling or a disease of addictive metabolic processes (e.g., Warburg effect), etc. We only focus on the cancer phenotypes and their dynamics, without an explicit need for causal mechanistic models (which may be governed by genomics, epigenomics, transcriptomics, proteomics, mateabolomics, etc).

2. Our models postulate finitely many *discrete phenotypes* that can be exhibited in cancer. The dynamics, possible transitions from one phenotype to another, are known, since they could be extracted by statistical analysis of patients, model animals, cancer cell lines or systems biology data. In the future, the models may be further extended to assume certain stress-hallmarks (e.g., certain characteristics of the tumor population, stroma or microenvironment) or other types of hallmarks.

3. Our models' dynamics assumes that cancer progression is primarily driven by a Darwinian *somatic evolution* (based only on mutation and selection). In other words, exhibited changes in phenotypes are determined by genotypic changes. Possible genotypic changes are determined by various processes operating on the genome, e.g., point mutations, translocations, amplifications, deletions, loss of heterozygosity (LOH), which collectively may be labeled as a "Genome Organizing Device" (GOD). We only assume that GOD creates diversity, but not how exactly GOD functions. Advantageous cancer phenotypes of hallmarks are successively 'selected'.

4. Our models further assume that each hallmark state has an associated (stochastic) *hitting time*, i.e., a particular instance of a *stopping time*, representing the first time the modeled process "hits" a successor hallmark – a well-defined subset of the state space. The stopping time is modeled using Fisher's Fundamental Theorem of Genetics, and is incorporated in the clocks of timed CHAs:

$$\frac{d}{dt}\langle F\rangle = \sigma^2 - \mu\Delta_\mu,$$

where F represents a fitness function (corresponding to a hallmark) mapping genotypes to phenotypes, with the diversity in genotypes determining the variability in fitness, whose density function is over the population, $\langle F\rangle$

and σ^2 are respectively the expectation and variance over the population and μ a mutation rate, and Δ_μ a contribution in average gain/loss in fitness due to mutation (which may be non-additive due to epistasis). A straight-forward derivation (with a simple model of genotypes), for the case involving mutation and selection (but no recombination) may be found in [35].

Note that in our framework we do not need to know how precisely the inter-hallmark clocks move, but only that they can be described by certain stochastic or ordinary differential equations, whose parameters are obtained from elsewhere. The fitness function may be assumed to change over time (e.g., an uncontrolled proliferative state may lead to hypoxia in a certain sub-population of cells and confer a higher fitness to mutations that promote angiogenesis, or other similar Malthusian effects), σ^2 may be non-constant (e.g., the tumor's clonality and progression rates being variable), and also the mutation rates may vary over time. Thus, all we take from this model is just the ability to represent the clocks mathematically, without being encumbered by the different mechanisms involved in different hallmarks.

Altogether, these assumptions lead to a model of cancer progression that uses the classical formalisms of *hybrid automata* with multiple clocks, whose mathematical and computational structures have been well-studied.

5 Cancer Hallmark Automata

A simple, intuitive example CHA is shown in Figure 1. It comprises the following hallmarks (see [17] for more details):

SSG: Self-sufficiency in growth signals. Roughly speaking, cells no longer depend on external growth-promoting signals, but grow autonomously. Usually, such a state is associated with a gain of function of an oncogene or a loss of function of a tumor suppressor gene.

IAG: Insensitivity to anti-growth signals. Cells with this hallmark continue to grow even in the presence of inhibiting signals. Usually, certain cell-cycle checkpoints are no longer properly regulated.

Ang: Sustained angiogenesis. This state enables a cancer cell to signal for the construction of blood vessels.

LRP: Limitless replicative potential. While most normal cells can only divide a certain number of times, cells with this hallmark can divide without limits. In this state, a cancer cell may upregulate telomerase to restore telomere lengths.

EvAp: Evading apoptosis. Normally, cells have a program for controlled cell-death, which is used to remove damaged or otherwise unwanted cells. This program is disabled in this hallmark which enables cells with highly corrupted DNA to survive, which facilitates cancer.

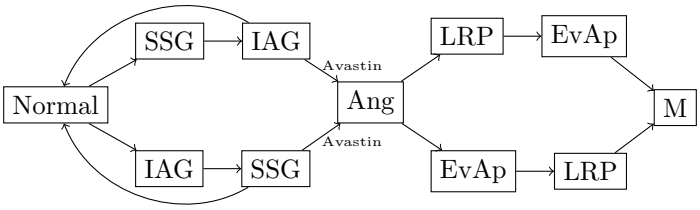

Figure 1: A simple hallmark automaton whose progression can be stalled by a VEGF-inhibitor such as Avastin.

M: Metastasis.

Various possible progressions through these hallmarks can be seen as transitions in the picture (note that this is a simplified and incomplete model). For example, Ang can be acquired after SSG and IAG. Moreover, as mentioned in Section 1, if a growing tumor fails to acquire Ang, it may starve; in this case, a solid tumor is unable to grow further and attain the later hallmarks. For simplicity, it may be modeled as a transition to the normal state.

In this example, the therapy "give the drug Avastin whenever a state leading up to Ang is reached" will prevent the cancer from reaching M.

5.1 Formal model

In the following, we start with a preliminary and simple formalization of the notions described above. We will successively extend the formal model in the later sections.

We assume a global set D of **drugs**.

Definition 5.1. A **Cancer Hallmark Automaton (CHA)** is a tuple

$$H = (V, E, v_0) \ ,$$

where

- V is a set of states, corresponding to hallmarks,[4]
- $E \subseteq V \times 2^{\mathcal{D}} \times V$ is a set of directed edges labeled with sets of drugs, and
- $v_0 \in V$ is the initial state.

We usually omit v_0 and write just (V, E).

Intuitively, an edge (v, D, v') represents a transition from state v to state v' that can be inhibited by any drug from the set $D \subseteq \mathcal{D}$. We allow several drugs to be given simultaneously and refer to such sets $C \subseteq \mathcal{D}$ of drugs as **cocktails**.

[4]Strictly speaking, a state corresponds to a subset of hallmarks that have been acquired.

Given a cocktail C, the edge $(v, D, v') \in E$ is *inhibited* by C if $C \cap D \neq \emptyset$. Given a state v and a cocktail C, v can transition to v' under C, in symbols $v \xrightarrow{C} v'$, if there is an edge (v, D, v') that is not inhibited by C. Note that we allow multiple edges (with different labels) between the same two states. To prevent a transition between two states, all edges connecting them need to be inhibited, which is why we need to consider cocktails rather than just single drugs. We assume that for every state v and every cocktail C there exists some state v' such that $v \xrightarrow{C} v'$ (possibly $v' = v$, these edges were omitted in Figure 1).

A **run** of a CHA $H = (V, E, v_0)$ is a sequence of transitions in E. Let $\mathit{Runs}(v, H)$ denote the set of runs that start in v. We write $\mathit{Runs}(H)$ for $\mathit{Runs}(v_0, H)$, and by $\mathit{Runs}_{\mathsf{f}}(v, H)$ we denote the set of finite runs from $\mathit{Runs}(v, H)$.

We now formalize how it is possible to *interfere* with the progression of the system.

Definition 5.2. A **therapy** is a function $\pi : \mathit{Runs}_{\mathsf{f}}(H) \to 2^{\mathcal{D}}$. A **possible execution** of π in H is a run

$$S = v_0 v_1 v_2 \dots \ ,$$

such that for each $i \geq 0$, $v_i \xrightarrow{\pi(S_i)} v_{i+1}$, where S_i denotes the initial segment of S up to step i.

Definition 5.3. Costs are given by the following (overloaded) function, for some finite dimension n:

- $c : V \to \mathbb{R}^n_{\geq 0}$ specifying costs of states,
- $c : 2^{\mathcal{D}} \to \mathbb{R}^n_{\geq 0}$ specifying costs of cocktails.

Thus, both states and cocktails have costs assigned to them, represented as n-dimensional vectors. Dimensions may include toxicity of the drugs, monetary cost of the drugs, discomfort for the patient, etc.

The cost of a possible execution $S = v_0 v_1 v_2 \dots$ of a therapy π with *discount factor* $0 < \delta \leq 1$ is

$$c(S, \pi, H) = \sum_{i \geq 0} \delta^i \big(c(v_i) + c(\pi(S_i))\big) \ .$$

The set of possible costs of π for a CHA H is

$$c(\pi, H) = \{c(S, \pi, H) \mid S \text{ is possible execution of } \pi \text{ in } H\}.$$

Now that we have a definition of the set of possible costs of a therapy, we can compare different therapies with respect to their costs.

Definition 5.4. A cost vector $x \in \mathbb{R}^n$ **Pareto-dominates** another vector $x' \in \mathbb{R}^n$, in symbols $x \prec x'$, iff for each $1 \leq \ell \leq n$ we have $x_\ell \leq x'_\ell$ and for some $1 \leq \ell \leq n$ we have $x_\ell < x'_\ell$.

A therapy π **Pareto-dominates** a therapy π' in a CHA H if for each $x \in c(\pi, H)$ and $x' \in c(\pi', H)$ we have $x \prec x'$. The set of **candidate therapies** for H is

$$\Theta(H) = \{\pi \mid \pi \text{ is not Pareto-dominated in } H\} \ .$$

Note. For the special case of 1-dimensional costs (or if there is a function to aggregate cost vectors into single numbers), the set of candidate therapies is the set of therapies whose best-case cost is not higher than some other therapy's worst-case cost.

This definition of a set of candidate therapies is a very conservative one, in that it includes any therapy that is not overtly worse than some other therapy. There are different possibilities for defining the set of candidate therapies, or for pruning the set further. Examples of such strategies for pruning the set further include *maximin*, i.e., choosing those strategies that lead to the best worst-case outcome, or *maximax*, i.e., choosing those strategies that lead to the best best-case outcome. However, making these decisions depends on the risk attitude of patient and doctor which may not be fully formalizable. Therefore we include all the potentially relevant therapies in the set of candidate therapies.

In order to be clinically applicable, a hallmark model may need to be *personalized* for any given patient or cancer type. Richer models in the future will necessitate more personalization. This personalization will result in families of hallmark automata, with different sets of candidate therapies. While we will not give full details here, we want to describe one possible application for such richer models.

For families of hallmark automata, we can ask whether there are any *universal* therapies for all of the included automata. Such therapies can result in faster and cheaper treatments.

To be able to apply therapies across different automata, their domain must be the same. This can be achieved, for example, by considering CHAs that contain the same set of hallmarks, and therapies that either depend only on the current state, or that have the set of all sequences of states as domain. The following definition applies to therapies on such unified domains.

Definition 5.5. Given a family $\mathcal{H}$ of hallmark automata, the set of **(universal) candidate therapies** for $\mathcal{H}$ is

$$\Theta(\mathcal{H}) = \bigcap_{H \in \mathcal{H}} \Theta(H) \ .$$

A set θ of therapies **covers** $\mathcal{H}$ if

$$\theta \cap \Theta(H) \neq \emptyset \text{ for all } H \in \mathcal{H} \ .$$

Note that if $\Theta(\mathcal{H}) \neq \emptyset$ then for each $\pi \in \Theta(\mathcal{H})$, $\{\pi\}$ covers $\mathcal{H}$.

5.2 Temporally extended goals: CTL

We have seen in the previous section that therapies can be compared according to their costs. Thus, the problem of finding the right therapy can be viewed as an optimization problem. It can, however, be necessary to have more detailed control over the therapeutic objectives. Simple reachability properties can be used as goals, such as "metastasis will never be reached". For more expressivity we can use Computation Tree Logic (CTL) [11] to specify goals.

Example 5.6. The goal $\mathsf{AG}\neg\mathrm{M}$ says that metastasis is never reached. Another possible goal could be

$$\mathsf{AG}(\mathrm{Ang} \to \mathsf{AG}\neg\mathrm{EvAp}) \ .$$

This means that whenever sustained angiogenesis is acquired, then at no point in the future the capability of evading apoptosis will be obtained.

One may be interested in checking properties of the CHA itself, without application of a therapy. This can be done using CTL model checking (see, e.g., [12]). CTL properties can also be checked on the possible executions of a given pair of therapy and hallmark automaton. Supervisory control for finite automata with CTL goals is known to be EXPTIME-complete, and controller synthesis algorithms exist [28].

The above representation of a cancer hallmark automaton is intuitive, but its simplicity also has some shortcomings. First, the above formalism does not include *timing*. Some transitions could be very short while others may take many years. A second limitation is that it does not model *uncertainty* about the current state. Typically, a clinician cannot know exactly how far progressed a cancer is, and must design therapies taking this uncertainty into account. Moreover, a clinician may decide to perform a *test* to reduce uncertainty. Third, cancer cannot be treated without considering the rest of the body. For example, it may be possible to prevent a cancer from evolving to metastasis but at the same time causing the liver to enter a highly toxic state.

We will address these issues in the remainder of this paper. In the next section we introduce timed CHAs, subsequently we incorporate tests and belief sets into the framework in Section 7, and finally we introduce an exemplary liver model and the concept of a product automaton in Section 8.

6 Timed CHAs

The framework we built so far is somewhat idealized in that transitions occur spontaneously and drugs can switch off transitions completely. More realistically, transitions would take certain durations of time, and drugs can slow down (or stop) the transition process. For example, in pancreatic cancer, it takes about a year for K-*ras* mutations in a cell to lead to neoplasms (so-called PanINs) [26]. To model durations, we will now add a notion of *time* to our CHA framework.

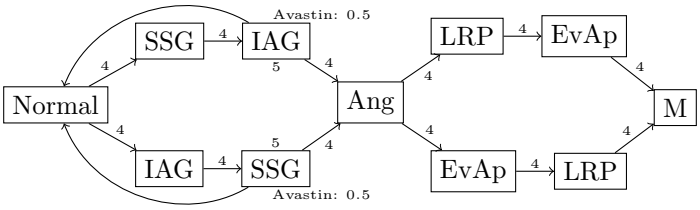

Figure 2: A simple timed hallmark automaton, using one clock (not named explicitly). The edges are labeled with the minimum times needed to make the respective transitions. In the two states that lead up to Angiogenesis, Avastin can be given to slow down the progress by a half. Those states are labeled with invariants, and depending on the precise timing, these invariants can force the system back to Normal before the transition to Angiogenesis is possible.

We start with the assumption that the acquisition of a hallmark requires a certain minimum amount of time. We do not specify exactly how that time is determined, but it could be the stopping time of a stochastic process such as randomizing over a set of driver mutations (see Section 4), or some value obtained from clinical data. Only after that time a given transition will be possible, and as mentioned, drugs can be used to prolong this time.

Further, we allow states to have *invariants*, specifying the maximum time that the system can remain in the respective state. For example, a tumor may only be able to remain in a state of unbounded growth without angiogenesis for a certain number of months.

Figure 2 shows the automaton from Figure 1 with timing information added, illustrating this intuition. We formalize the extension in the following.

We assume a finite set X of real-valued variables called *clocks*, over which the set of *constraints* $\mathcal{C}(X)$ is generated according to the grammar

$$\phi ::= x \geq k \mid \phi \wedge \phi \ ,$$

where $k \in \mathbb{N}$ and $x \in X$. A *valuation* of the variables in X is a mapping $\mathsf{val} : X \to \mathbb{R}_{\geq 0}$. We denote the null valuation $x \mapsto 0$ by 0. By $\mathsf{val} \models \phi$ we denote that val satisfies ϕ.

Definition 6.1. A **timed CHA** is a tuple $H = (V, E, v_0, \ell, \rho)$ where

- V is a set of states, corresponding to hallmarks,
- $E \subseteq V \times \mathcal{C}(X) \times V$ is a set of directed edges each labeled with a clock constraint,
- $v_0 \in V$ is the initial state,
- $\ell : V \times X \to \mathbb{N}$ is a partial function specifying the time limit (if any) for each clock that the system can remain in a given state (this is also called the *invariant*), and

- $\rho : V \times \mathcal{D} \times X \to \mathbb{R}_{\geq 0}$ yields a function specifying how a given drug influences the clocks at a given state.

Intuitively, at a given state v, the drug d slows down or speeds up the clock x as specified by the factor $\rho(v, d, x)$. If the factor is 1 the drug has no effect on that clock, and if it is 0 it effectively stops the clock from progressing. If several drugs have an effect on a clock, their factors are multiplied. We extend ρ to cocktails by setting $\rho(v, C, x) = \prod_{d \in C} \rho(v, d, x)$ for any cocktail $C \neq \emptyset$, and $\rho(w, \emptyset, x) = 1$.

A directed edge (v, ϕ, v') represents a transition from v to v' that can take place once the time constraint ϕ is satisfied.

We assume that for each state v that has a time limit for a clock x, there is an outgoing edge (v, ϕ, v') such that $\mathsf{val} \models \phi$ for all val with $\mathsf{val}(x) = \ell(v, x)$.[5] This edge specifies the behavior of the system if the respective clock reaches its time limit.

The cost functions in the context of timed CHAs are the same as those for the untimed version, but with a timed interpretation: $c(v)$ is the cost of staying at state v per time unit (days/weeks/months/years), and $c(C)$ is the cost of administering a drug cocktail C per time unit.

We next see how to adapt the definitions related to runs of a CHA to the timed version, starting with the notion of a *timed state*.

Definition 6.2. A **timed state** of a timed CHA (V, E) is a tuple $(v, \mathsf{val}) \in V \times \mathbb{R}^X$, where v is a hallmark state and val a clock valuation. There are two types of **transitions** between timed states:

1. *Delay* transitions, in symbols $(v, \mathsf{val}) \xrightarrow{\delta, C} (v, \mathsf{val}')$, where
   - $\delta \in \mathbb{R}_{>0}$ represents the (real) time delay,
   - C denotes the cocktail active during that time,
   - $\mathsf{val}'(x) = \mathsf{val}(x) + \delta\rho(v, C, x)$ for all x, and
   - $\mathsf{val}'(x) \leq \ell(v, x)$ for all x with $\ell(v, x)$ defined.
2. *State* transitions, in symbols $(v, \mathsf{val}) \to (v', 0)$, where
   - there is an edge $(v, \phi, v') \in E$ with $\mathsf{val} \models \phi$

Note that whenever a state transition takes place, the clocks are reset. This is to simplify our presentation and could be replaced by explicit clock resets as common in the literature.

This setup includes the special case where there is one clock unaffected by any drug, representing real time. Invariants over that clock can be used to specify, for example, how many months the tumor can remain in a certain hallmark state.

[5] Note that this requires $\mathsf{val} \models \phi$ even for valuations that exceed some other clock's invariant; however, this does not have an effect since we only allow $\geq$ constraints on the edges.

This timed setup can also emulate the concept of edges labeled with drugs that inhibit them. This can be achieved as follows: Suppose we want to model an edge between two states v, v' that can be inhibited by a drug d. Then we can introduce a clock variable $x_{d,v'}$ with $\rho(v, d, x_{d,v'}) = 0$, and add a constraint $x_{d,v'} \geq z$ to the edge between v and v', for some $z > 0$. As long as drug d is given before the constraint is satisfied, the transition will be inhibited. However, once the constraint is satisfied, the tumor has advanced too far and it is no longer possible to inhibit the transition.

A *run* in the case of a timed CHA H is a non-Zeno[6] sequence of delay and state transitions. Similar as before, let $\mathit{Runs}((v, \mathsf{val}), H)$ denote the set of runs that start in (v, val). We write $\mathit{Runs}(H)$ for the set $\mathit{Runs}((v_0, 0), H)$, and $\mathit{Runs}_{\mathsf{f}}((v, \mathsf{val}), H)$ for the set of finite runs from $\mathit{Runs}((v, \mathsf{val}), H)$.

Definition 6.3. A **therapy** is a function $\pi : \mathit{Runs}_{\mathsf{f}}(H) \to 2^{\mathcal{D}}$. A **possible execution** of π in H is a run

$$S = (v_0, 0)(v_1, \mathsf{val}_1)(v_2, \mathsf{val}_2) \dots$$

such that for all i with delay transitions $(v_i, \mathsf{val}_i) \xrightarrow{\delta, C} (v_{i+1}, \mathsf{val}_{i+1})$,[7] for every $0 \leq \delta' < \delta$

$$\pi((v_0, 0) \dots (v_i, \mathsf{val}_i)(v_i, \mathsf{val}_i + \delta' \rho(v_i, C))) = C \ ,$$

where $\rho(v_i, C)$ denotes the partial evaluation of ρ, i.e., the function $x \mapsto \rho(v_i, C, x)$.

This last condition ensures that the therapy does not change during a transition, or, put differently, that a change in therapy is always reflected by starting a new transition.

For any finite run $r \in \mathit{Runs}_{\mathsf{f}}(H)$, we denote its *duration* as

$$\tau(r) = \sum_{0 \leq j < len(r)} \begin{cases} \delta & \text{if } r_j \xrightarrow{\delta, C} r_{j+1} \text{ for some } \delta, C \\ 0 & \text{otherwise,} \end{cases}$$

where $len(r)$ denotes the length of the state sequence in r and r_i its initial segment of length i.

Definition 6.4. Given a CHA H and a possible execution S of a therapy π, the **cost** of S given π with discount factor $0 < d \leq 1$ is

$$c(S, \pi, H) = \sum_{i \geq 0} \frac{1}{d} \left(e^{-d\tau(S_i)} - e^{-d\tau(S_{i+1})} \right) (c(v_i) + c(\pi(S_i)))$$

(as before, by S_i we denote the initial segment of S up to step i). This simple discounting function does not necessarily capture a real patient's preferences, but

[6] That is, not containing an infinite chain of timed transitions with convergent total duration.
[7] Note that $v_i = v_{i+1}$.

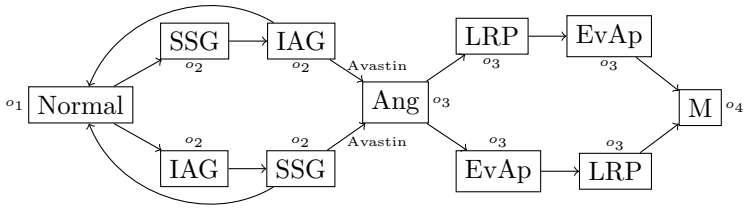

Figure 3: A simple hallmark automaton with test observations $o_1, \ldots, o_4$.

any convergent function will work in its stead. We will consider more realistic functions in the future, which can potentially be designed on a case-by-case basis depending on the patient's valuation.

The set of possible costs of π in a timed CHA H is the set of costs of possible executions of π,

$$c(\pi, H) = \{c(S, \pi, H) \mid S \text{ is possible execution of } \pi \text{ in } H\} \ .$$

The notions of Pareto dominance and universal therapies carry over from untimed CHAs.

6.1 Timed CTL

We can extend the CTL goals of the previous section to include time [1]. For example, the goal $\mathsf{AG}_{\leq 20}\neg\mathrm{M}$ says that metastasis is not reached within 20 time units (e.g., 20 years). This kind of goal represents the approach of turning cancer into a chronic disease, rather than trying to cure it completely. For example, the above formula may be appropriate for a patient of sixty years of age, who may then be able to get a less strenuous therapy, while for a younger patient the time requirements may be more extensive.

Out of all the therapies satisfying a CTL goal, the best ones may be chosen either by a separate cost optimization, or by incorporating cost requirements into the formulas using a weighted version of CTL [7].

7 Partial observability and tests

So far, we assumed perfect information about the state of the system. In reality however, a clinician will only have partial observations of the tumor's internal state. To reduce uncertainty about the current state of the cancer progression, tests can be performed. In this section, we extend our formal framework to include partial observability and tests, both for untimed and timed CHAs.

7.1 Tests in untimed CHAs

We view tests as functions mapping states to observations.[8] See Figure 3 for an example of such a test with 4 possible observations. When the test yields observation o_2, we know that the system is in a state prior to acquiring sustained angiogenesis, and that we can give a VEGF-inhibitor such as Avastin to inhibit the progression to a hallmark promoting construction of new blood vessels to the tumor. A more fine-grained test, or another test with intersecting observations, would have to be performed to determine the state more precisely, e.g., whether it is in the upper or in the lower branch of the automaton, and thus whether other potential drugs should be preferred.

Formally, for a CHA (V, E) we assume a set T of **tests** and a set O of **observations**. Each test $t \in T$ is a function $t : V \to O$, inducing a partition on the set of states. When performing test t while the system is in state v, the resulting observation allows the conclusion that the system is in one of the states in the equivalence class of v with respect to that partition.

We now extend the notion of therapy to include tests. We assume that tests only acquire information, without affecting the state of the system. That is, given a test t and a state v the system can only *transition* to v itself: $v \xrightarrow{t} v$.

We can keep track of the information that results from tests by adding belief sets to runs. A belief set is a subset of states that the system may be in at a given moment. We augment states with belief sets to obtain *belief states*.

Definition 7.1. A **belief state** of a CHA (V, E) is a tuple (v, b), where

- $v \in V$ a state,
- $b \subseteq V$ with $v \in b$ is a belief set.[9]

There is a *transition* from belief state (v, b) to (v', b') under $a \in 2^{\mathcal{D}} \cup T$ if

- $v \xrightarrow{a} v'$ and
- $b' = \begin{cases} [b]_{\xrightarrow{C}} & \text{if } a = C \in 2^{\mathcal{D}} \\ \{v' \in b \mid t(v') = t(v)\} & \text{if } a = t \in T \end{cases}$

where $[X]_R$ denotes the image of set X under relation R, i.e., $[X]_R = \{x' \mid (x, x') \in R \text{ with } x \in X\}$.

In symbols, we write $(v, b) \xrightarrow{a} (v', b')$. In addition to an initial state v_0, we now also have an initial belief set b_0. So a CHA is now a tuple (V, E, v_0, b_0), and

[8] The test we describe here are deterministic, i.e., for any given state, a certain test always leads to the same observation. In the literature, non-deterministic tests are common, where a test may lead to one of a set of possible observations. Our framework can be extended in the same way, but from the biological perspective, that would mean that there are different mechanistic causes for the system being in that state. In that case, we recommend refining the model to have different states representing these different causes.

[9] Note that belief states correspond to *pointed models* in epistemic logic, in the sense that they consist of a set of *possible* states with a distinguished *actual* one.

a **run** of a CHA H is now a sequence of transitions over belief states. As before, $\mathit{Runs}((v,b),H)$ denotes the set of runs that start in (v,b). We write $\mathit{Runs}(H)$ for $\mathit{Runs}((v_0,b_0),H)$, and by $\mathit{Runs}_{\mathsf{f}}((v,b),H)$ we denote the set of finite runs from $\mathit{Runs}((v,b),H)$.

We now extend the notions of therapies and their execution to include tests and belief sets.

Definition 7.2. A **therapy** is a function $\pi : \mathit{Runs}_{\mathsf{f}}(H) \to 2^{\mathcal{D}} \cup T$. It is **uniform** if it only depends on the belief sets.[10] We only consider uniform therapies, without explicitly mentioning it.

A **possible execution** of π in H starting with (v_0,b_0) is a run

$$S = (v_0,b_0)(v_1,b_1)(v_2,b_2)\dots \ ,$$

such that for each $i \geq 0$, $(v_i,b_i) \xrightarrow{\pi(S_i)} (v_{i+1},b_{i+1})$.

We also extend the definition of costs, using $c : T \to \mathbb{R}^n_{\geq 0}$ to specify costs of tests. The definition of cost of an execution then is the same as in Definition 5.3, and we can proceed with the notion of possible costs.

Definition 7.3. The set of possible costs of π for a CHA H is

$$c(\pi,H) = \{c(S,\pi,H) \mid S \text{ is a possible execution of } \pi \text{ in } H \text{ starting with } (v,b_0) \text{ for any } v \in b_0\}.$$

The remaining notions such as Pareto dominance, candidate therapies, and universal therapies remain unchanged.

7.2 Epistemic and Temporally extended goals

Given that we now have a framework that captures not only the actual behavior of the system but also the observer's (e.g., oncologist's) information about it, we need to reflect this additional aspect in the formal language that defines goals. This can be done by adding an epistemic modality K to the logic, which intuitively means "it is known that".

Instead of the previously mentioned goal $\mathsf{AG}\neg\mathrm{M}$, we can now express that it is known that metastasis is never reached by stating $K\mathsf{AG}\neg\mathrm{M}$.

Another, somewhat more complex, goal is

$$\mathsf{AG}\big(\mathrm{Ang} \to \big((\neg\mathrm{M} \wedge \mathsf{AX}\neg\mathrm{M})\,\mathsf{U}\,K\mathrm{Ang}\big)\big)$$

which intuitively says that whenever the tumor acquires angiogenesis, this will be known (strictly) before the tumor reaches metastasis.[11] Any such goal formula should implicitly be put inside an enclosing K operator to ensure that it holds in all starting states initially considered possible.

[10] More precisely, if for any two runs $r = (v_0,b_0)(v_1,b_1)\dots(v_k,b_k)$ and $r' = (v'_0,b_0)(v'_1,b_1)\dots(v'_k,b_k)$ which agree on their belief sets, we have $\pi(r) = \pi(r')$.

[11] More precisely, the statement is that at any point in the future where Ang holds, M will not hold at the current or the next step until Ang is known (where Ang is the Angiogenesis hallmark and M the Metastasis hallmark).

7.3 Tests in timed CHAs

Analogously to untimed CHAs, we also extend the timed CHA framework to include belief sets and tests. A *belief set* b now is not just a set of states v, but a set of timed states (v, val). A *belief state* is a tuple (v, val, b) such that $(v, \mathsf{val}) \in b$. As before, we assume some initial belief set b_0 that is used when no other belief set is given.

Before we generalize the notions of transitions and executions of a therapy we need to introduce a new relation. It addresses the following issue: With full observability, we can identify the individual delay or state transitions; however, with partial observability, a sequence of several transitions might look like just one transition to the outside observer. We denote such multi-step transitions using $\overset{\delta,C}{\dashrightarrow}$, which relates any two states that are related by any number of transitions under C taking a total time of δ. Formally, for two timed states (v, val) and (v', val'), we have $(v, \mathsf{val}) \overset{\delta,C}{\dashrightarrow} (v', \mathsf{val}')$ if there exists a sequence

$$S = (v, \mathsf{val})(v_1, \mathsf{val}_1) \dots (v_k, \mathsf{val}_k)(v', \mathsf{val}')$$

of state or delay transitions under C, with $\tau(S) = \delta$. (Recall that $\tau(S)$ denotes the total duration of execution S.)

Definition 7.4. In timed CHAs with partial observability, there are three types of **transitions** between belief states:

1. *Delay* transitions, in symbols $(v, \mathsf{val}, b) \xrightarrow{\delta,C} (v, \mathsf{val}', b')$, where
   - $(v, \mathsf{val}) \xrightarrow{\delta,C} (v, \mathsf{val}')$ and
   - $b' = [b]_{\overset{\delta,C}{\dashrightarrow}}$
2. *State* transitions, in symbols $(v, \mathsf{val}, b) \to (v', 0, b')$, where
   - $(v, \mathsf{val}) \to (v', 0)$ and
   - $b' = [b]_{\overset{0,C}{\dashrightarrow}}$, that is, all state transitions under C
3. *Test* transitions, in symbols $(v, \mathsf{val}, b) \xrightarrow{t} (v, \mathsf{val}, b')$, where
   - $b' = \{(v', \mathsf{val}') \in b \mid v' \in t(v)\}$.

Note that tests in this formulation only give information about the current state, and not about the current clock values. If deemed biologically plausible, this can be extended appropriately.

Note also that test transitions are assumed to be instantaneous. We make this assumption because receiving the result of a test usually takes hours or days, whereas tumors usually progress on a larger time scale (months or years).

As before, a *run* of a timed CHA H with tests is a non-Zeno sequence of delay, state and test transitions.

Definition 7.5. A **therapy** is a function $\pi : \mathit{Runs}_{\mathsf{f}}(H) \to 2^{\mathcal{D}} \cup T$. Again, a therapy is **uniform** if it only depends on the belief sets, and we only consider uniform therapies, without explicitly mentioning it. A **possible execution** of π in H is a run

$$S = (v_0, 0, b_0)(v_1, \mathsf{val}_1, b_1)(v_2, \mathsf{val}_2, b_2) \dots$$

such that

- for all i with delay transition $(v_i, \mathsf{val}_i, b_i) \xrightarrow{\delta, C} (v_{i+1}, \mathsf{val}_{i+1}, b_{i+1})$ and for every $0 \leq \delta' < \delta$,

  $$\pi((v_0, 0, b_0) \dots (v_i, \mathsf{val}_i, b_i)(v_i, \mathsf{val}_i + \delta' \rho(v_i, C), [b_i]_{\dashrightarrow}^{\delta', C})) = C \ ,$$

  where $\rho(v_i, C)$ denotes the partial evaluation of ρ, i.e., the function $x \mapsto \rho(v_i, C, x)$, and

- for all i with test transition $(v_i, \mathsf{val}_i, b_i) \xrightarrow{t} (v_{i+1}, \mathsf{val}_{i+1}, b_{i+1})$,

  $$\pi((v_0, 0, b_0) \dots (v_i, \mathsf{val}_i, b_i)) = t \ .$$

The definition of costs is analogous to Definition 6.4, except that tests have to be treated separately since they take no time (and would thus add no costs). The formula can straightforwardly be modified to count the costs of tests at some constant rate (still discounting the future).

Again, the notions of cost of executions, Pareto dominance, universal therapies, non-Zeno-ness and null therapies are the same or very similar as with untimed CHAs.

7.4 Therapies as conditional plans

In this section, we show how a therapy can be interpreted as a conditional plan instead of a function from runs to actions. Intuitively, a conditional therapy plan is a sequence of therapeutic actions, which branches after each test action into distinct sub-cases according to the possible observations of the test. We give the formal translation of a therapy π into a conditional plan π_c below.

Before we proceed, we note that, due to uniformity, a therapy can be regarded as a function assigning actions to sequences of belief sets (rather than executions). We write b_S for the sequence of belief sets in S. When S is clear from the context, we drop the subscript and simply write b. By $\mathrm{b} \circ b$ we denote the sequence b with belief set b appended.

Definition 7.6. Given a sequence of belief sets $\mathrm{b} = b_0 \dots b_n$, a time τ and a therapy π we define a conditional plan π_c as follows:

- If $\pi(\mathrm{b}) = C \in 2^{\mathcal{D}}$, then

  $$\pi_c(\mathrm{b}, \tau, \pi) = (C, \tau); \pi_c(\mathrm{b} \circ [b_n]_{\dashrightarrow}^{\delta, C}, \tau + \delta, \pi) \ ,$$

  where δ is the minimum value such that

– $\pi(\mathrm{b} \circ [b_n]_{\xrightarrow{\delta, C}}) \neq C$, and

– $\pi(\mathrm{b} \circ [b_n]_{\xrightarrow{\delta', C}}) = C$ for all δ' such that $0 \leq \delta' < \delta$.

- If $\pi(\mathrm{b}) = t \in T$ with possible observations $o_1, \ldots, o_k$, then

$$\pi_c(\mathrm{b}, \tau, \pi) = (t, \tau); \mathtt{case} \begin{bmatrix} o_1 : \pi_c(\mathrm{b} \circ (b_n \cap O_1), \tau, \pi) \\ \ldots \\ o_k : \pi_c(\mathrm{b} \circ (b_n \cap O_k), \tau, \pi) \end{bmatrix}$$

  where $O_i = \{(v, \mathsf{val}) \in V \times \mathbb{R}^X_{\geq 0} \mid t(v) = o_i\}$, and the `case` statement has the intuitive meaning, as explained below.

Given the initial belief set b_0, the conditional plan that corresponds to the therapy π is defined as $\pi_c(b_0, 0, \pi)$.

The intuition behind this translation is as follows. Since a therapy only depends on the sequence of belief sets, and the evolution of belief sets under any cocktail C is predetermined, we can compute when the therapy will change. For example, starting at the initial belief set b_0 with initial cocktail C, the therapy changes at the smallest δ such that $\pi([b_0]_{\xrightarrow{\delta, C}}) = C'$ for some $C' \neq C$. The new belief set at this moment is $b_1 = [b_0]_{\xrightarrow{\delta, C}}$, and the conditional plan up to this point is $(C, 0); (C', \delta)$. We can continue this procedure with the sequence $b_0 b_1$. When a test is performed, the next move depends on the observation o_i, which is reflected in the branching `case` statement. The execution of such a therapy plan would then continue at the branch labeled with the observation.

8 Liver and Product Automata

In a patient, cancer itself is not the only system of relevance. Other systems interact with the tumor's development, and especially during a therapeutic intervention, they need to be monitored. For example, the immune system and its role throughout carcinogenesis are receiving more and more attention [14], and the liver needs to be monitored to avoid damage due to excess toxicity.

In principle, other subsystems of an organism could be modeled as hybrid automata in the same way as our CHA, which could then be composed to an overall model for which therapies with goals spanning all subsystems could be generated. We postpone a discussion of the general framework and sketch here only a simple toxicity-based liver model that can be "attached" to a CHA. It has only one clock, modeling one type of toxicity level, and a very simple discrete dynamics governed by a sequence of thresholds. Simple as it is, this kind of model can still capture effects that are discussed in the literature, such as the dynamics of the toxicity level in the liver caused by Taxol [40], a drug used in breast cancer treatment.

Definition 8.1. A **liver automaton** is a tuple $L = (W, F, w_0, \ell, \rho)$, where

- W is a set of states,
- $F \subseteq W \times W$ is a set of directed edges,
- $\ell : W \to \mathbb{R}$ gives the toxicity threshold for each state, and
- $\rho : W \times \mathcal{D} \to \mathbb{R}_{\geq 1}$ gives the toxicity factor for each pair of state and drug.

For simplicity, we restrict attention to *linear* liver automata, i.e., each state has at most one successor. For this reason, we do not need explicit constraints on the edges and can instead assume that a state's outgoing edge is enabled exactly when its toxicity threshold is reached.

We can then define the overall toxicity factor of a given cocktail in a given state as a function $\rho : W \times 2^{\mathcal{D}} \to \mathbb{R}$ as follows:

$$\rho(w, C) = \begin{cases} \prod_{d \in C} \rho(w, d) & \text{if } C \neq \emptyset \\ -1 & \text{if } C = \emptyset \end{cases}$$

Note that $\rho(w, \emptyset) = -1$, while for any $C \neq \emptyset$, we have $\rho(w, C) \geq 1$. That is, we assume that drugs cumulatively increase the toxicity level, and that the liver regenerates only when no drugs are given. The model can easily be extended to include some drugs that have no effect on the liver, or to allow for other interactions between cocktails.

Definition 8.2. A **timed state** of a timed liver automaton $L = (W, F, w_0, \ell, \rho)$ is a tuple (w, c), where $w \in W$ is a current state and $c \in \mathbb{R}$ is a current clock value for w.

There are three types of transitions between timed states in a liver automaton:

1. *Delay* transitions, in symbols $(w, c) \xrightarrow{\delta, C} (w, c')$, where
   - $\delta \in \mathbb{R}_{>0}$ represents the (real) time delay,
   - C denotes the cocktail active during that time,
   - $c' = \begin{cases} \max\{0, c + \delta\rho(w, C)\} & \text{if } w = w_0, \text{ and} \\ c + \delta\rho(w, C) & \text{otherwise} \end{cases}$
   - $-1 \leq c' \leq \ell(w)$.
2. *State* transitions, in symbols $(w, c) \to (w', 0)$, where
   - $c = \ell(w)$,
   - $(w, w') \in F$.
3. *Regenerating* transitions, in symbols $(w, -1) \to (w', c')$, where
   - $c' = 0$,
   - $(w', w) \in F$.

The exact thresholds for regenerating transitions can be modeled in more detail where required.

A liver automaton can be combined with a CHA using parallel composition as usual in automata theory [19]. We can then formulate combined goals about a CHA and the liver. For example, a goal might be to avoid a high level of toxicity (T_5) in a somewhat advanced stage of the progression (Ang):

$$\mathsf{AG}(\text{Ang} \rightarrow \neg T_5) \ .$$

9 Algorithms

Our focus in this paper has been on motivating and establishing a formalism that captures the biology behind cancer progression models. The technical and algorithmic properties of that formalism are rather involved, so we here focus on a brief initial assessment and provide a roadmap for the future.

Controller synthesis is hard even for untimed systems (e.g., EXPTIME-complete when using CTL goal specifications [28]), so high complexity classes are to be expected in our richer framework as well.

The controller synthesis problem for hybrid automata has been studied in the literature, however, often restricted to fully observable automata and to achieving *safety* properties. Such properties form a sub-class of what can be expressed in richer temporal logics such as CTL in that they only talk about avoiding certain bad states at all times. Such safety properties is relevant for CHAs, because goals such as "metastasis will never be reached" can be expressed. In the following, we will discuss how existing algorithms can be built on and adapted to handle CHAs, which differ somewhat from standard hybrid automata.

From timed CHAs to hybrid automata In the hybrid automata literature, the rates of the clocks are constant at any given state,[12] and what is controllable are (some of) the transitions between states. In our framework, in contrast, the rates of the clocks is what can be affected by control actions (drugs), while the transitions (tumor progression) cannot be directly manipulated. However, this difference is mainly conceptual, and we can translate CHAs to standard hybrid automata as follows, thus transferring existing results naturally.

Given a set of drugs $\mathcal{D}$ and a CHA with states V, we construct a hybrid automaton H in the following way: For each state $v \in V$ and each cocktail $C \in 2^{\mathcal{D}}$, H contains a state v_C with the same clock invariants as v. For any edge between two states $v, v' \in V$, H contains an uncontrollable edge between v_C and v'_C, for each cocktail C, with the same clock constraints and resets as on the CHA edge. In addition to the uncontrollable edges, there are controllable directed edges from v_C to $v_{C'}$ for each v, C, and C'. These edges represent changes of therapies, and have no clock constraints or resets. At a state v_C, the rate of each clock $x \in X$ is fixed, given by $\rho(v, C, x)$. The clock rates can be learned from patient data or mechanistically built on stochastic diffusion

[12] One exception are so-called *differential games* [32], which we plan to explore in the future.

equations (SDEs) based on Fisher's theorem, as motivated in Section 4. The precise methods will be discussed in a future paper.

This translation yields a hybrid automaton of size exponential in the number of drugs, but linear in the number of CHA states.

Undecidability of rectangular hybrid automata control It can be seen that the resulting hybrid automata are related to an important subclass of "simple" hybrid automata called *rectangular* hybrid automata, whose clocks' rates are defined using only upper and lower bounds.[13] Rectangular hybrid automata are an important subclass of hybrid automata because they have nice computational properties (e.g., decidability). For example, see [22], where the control problem for a rectangular game and an LTL formula is shown to be EXPTIME-complete in the size of the game. However, this complexity holds only for those automata that satisfy *initialization*, meaning that whenever a transition changes the activity of a variable, the value of the variable is reinitialized. In fact, for rectangular hybrid automata without initialization, even the reachability problem is undecidable [21]. Unfortunately, our CHA is not initialized, as we keep the clock values along controllable (drug) transitions while changing the rates of the clocks. We plan to further investigate the exact relationship between CHAs and rectangular hybrid automata in our future work, especially the case where clocks can be linked to clinical tests, or discretized and/or upper-bounded, as would be the case in all realistic applications.

Decidability of discrete-time CHA control The simplest way around the undecidability of the hybrid automata control problem is to allow for control moves (in our case, therapeutic interventions) only at discrete instants of time. Henzinger and Kopke [20] give an exponential-time algorithm for discrete-time safety control of rectangular hybrid automata with bounded and non-decreasing variables. They also show the problem to be EXPTIME-hard and discrete-time verification (CTL model checking) of rectangular hybrid automata to be solvable in PSPACE.

Even though our definition of timed CHAs does not require clocks to be bounded, such a restriction would not impose a severe limitation. By bounding the clocks by some value that even the healthiest patient will never reach, we can thus aim for decidability without forfeiting any meaningful therapy. The algorithms from [20] do not directly apply to CHAs as their framework requires all discrete transitions to be controllable, whereas our cancer progression transitions are uncontrollable. We plan to extend the algorithms from [20] to CHAs.

Note that even with discrete-time control, the decidability results described earlier no longer apply as we depart from rectangular hybrid automata. For example, in triangular automata (where clock constraints may be triangular predicates), even the safety verification problem is undecidable [20].

[13] In fact, they are in an even simpler class called *singular* hybrid automata, but rectangular automata are more extensively studied in the literature.

Initialized approximations of CHAs Another way of ensuring decidability is by modelling the "belief automaton" explicitly, so that instead of controlling the underlying cancer progression model, control occurs only on the belief level. The modified CHA can be implemented by assuming that tests not only give information about the current state of the system, but also give some bound on how long the system has already been in this state. That is, at a given state, a test has a discrete number of possible outputs. Now, if we require that every control action (change of cocktail) is preceded by a test action, all clocks can be set to the constants given by the test result and we obtain an initialized automaton.

The resulting automaton is still a rectangular hybrid automaton and hence algorithms from [21] generalize to the new model. In particular, the problem of safety control becomes decidable. In a sequel we will discuss the details of this approach.

Future work By expanding on these algorithmic considerations we will obtain a full treatment of the controller synthesis problem for both fully and partially observable CHAs. We can then specify goals formulated in CTL and the extensions mentioned in this paper, and also include cost optimization. Further issues include the characterization of universal therapies, that is, general therapies which "work well" for a whole class of patients or cancers (represented as parametrized CHAs).

10 Conclusions

This paper establishes a general formalism for describing cancer hallmarks and their dynamics, without relying on any detailed mechanistic model of cancer pathways (which can be included independently). Our goal was to design a conceptually clear framework based on realistic biological foundations.

We discuss below how our framework can be used, as is, to model phenomena beyond what we discussed so far, briefly illustrating two important cases. Then, we point out the limitations of the current paper and give a list of topics that we plan to approach next.

10.1 Modeling heterogeneity and anti-hallmarks

Heterogeneous tumors So far we have modeled states of a CHA as representing the unique dominant phenotype of the tumor cell population. However, most forms of cancer are not likely to be monoclonal, i.e., consist of only one population in which the clonal expansions postulated by Hanahan and Weinberg take place, but rather involve several sub-populations of tumor cells [34], each with a distinct dominant phenotype [16, 23]. In order to model this heterogeneity, we can simply think of a CHA state as representing a *vector* of dominant phenotypes, one for each sub-population. One or several components of such a vector may differ from one state to the next, corresponding to a change of

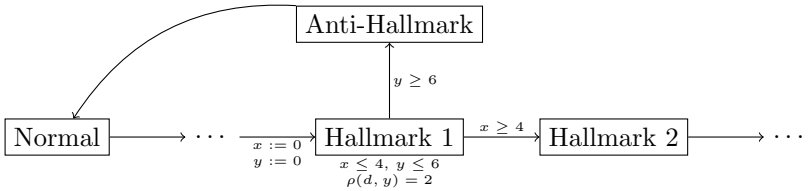

Figure 4: Illustrating how to model an anti-hallmark using two clocks x and y and a drug d that speeds up clock y at Hallmark 1 by a factor of 2.

the dominant phenotype in the corresponding sub-population(s) during the respective transition; or the length of the vector may change, corresponding to new distinct sub-populations emerging or existing sub-populations dying out. The arguments for our fundamental assumptions in Section 4 still apply, and it is not necessary to extend our formal CHA framework, since we view states as abstract entities without making any formal assumptions about their structure. This approach is, however, rather crude in modeling tumor heterogeneity, and does not straightforwardly accommodate, for example, information about tumor geometry or a model of the resulting spatial effects.

Anti-hallmarks Instead of trying to *slow down* cancer progression, there has recently been growing interest in approaches to *speed up* the process to a degree which will make the tumor inviable and "push it over the edge" towards collapse. We refer to such inviable states as *anti-hallmarks*. They can be modeled by putting constraints on the transitions leading to them that will never be satisfied, unless a drug is given which speeds up a certain clock. For example, consider the CHA in Figure 4. At Hallmark 1, without interference (both clocks increase with rate 1), the transition to Hallmark 2 will be taken after 4 time units. A drug that speeds up clock y by a factor of 2 will instead push the tumor to the Anti-Hallmark state, if given starting at most 1 time unit after entering Hallmark 1.

10.2 Future Work

Building on our conceptual foundation, we plan to address several important issues next.

Algorithmic issues We have only taken a preliminary look at algorithmic issues in Section 9, and plan to shift our focus to the algorithmic side of verifying cancer hallmark automata, automatically generating therapies, finding promising drug targets, etc.

Compositional models We illustrated the idea of compositional models comprising relevant subsystem of a tumor's host organism using a simple liver model in Section 8, and noted that its further generalization is likely to be

straightforward. Thus, we have omitted a full treatment of product automata, as well as additional complexities, e.g., various issues related to cellular stress, heterotypic interactions and metabolic processes involved in bio-energetics and bio-synthesis. These will be addressed in the future.

Model extraction Finally, we omitted a description of the methodologies needed for extracting cancer hallmarks and their temporal progression models from data or mechanistic pathway and population models. Currently, there is no consensus that the cancer hallmarks described in the literature constitute a complete list, nor is there a clear understanding (either phenomenologically or mechanistically) of their precise discrete dynamics. We also believe that spatial structure (geometry, growth curve, spatial distribution of heterogeneity, etc.) as well as motility (self-seeding, circulating tumor cells) may hold additional and important clues that can be easily incorporated into our therapy design [13, 36]. Therefore, we plan to extract models from spatio-temporal data, for example, data obtained from detailed simulations, or gene expression and imaging data from patients or mouse models. We plan to use statistical inference algorithms for model extraction in order to reconstruct temporal (or spatio-temporal) phenomenological models of cancer-related processes from such data.

GOALIE [41] is a recent system that has been used to reconstruct yeast metabolic and cell cycles from similar kinds of data, however, it only discovers a single path of behavior and it requires temporally annotated input data. Much of the patient data available (e.g., TCGA[14]) does not have sufficient temporal information, which therefore has to be inferred separately.

There exist algorithms that combine the temporal reconstruction with inference of trees of possible copy number event sequences in cancer [15, 42], yielding structures very similar to our CHAs. Since the time of writing of those works, more advanced technologies have been developed, so it will be necessary to adjust those models to handle the more fine-grained information available nowadays. We are also working on adapting specialized algorithms [27, 5] for separating relevant from irrelevant events, in order to refine the pre-processing steps of the progression reconstruction algorithms.

References

[1] R. Alur, C. Courcoubetis, and D. Dill. Model-checking in dense real-time. *Information and Computation*, 104(1):2 – 34, 1993. ISSN 0890-5401. doi: 10.1006/inco.1993.1024. URL `http://www.sciencedirect.com/science/article/pii/S0890540183710242`.

[2] Rajeev Alur. Timed automata. In Nicolas Halbwachs and Doron Peled, editors, *Computer Aided Verification*, volume 1633 of *Lecture Notes in Computer Science*, pages 688–688. Springer Berlin / Heidelberg, 1999.

[14] The Cancer Genome Atlas, `http://cancergenome.nih.gov/`

[3] Eugene Asarin, Oded Maler, Amir Pnueli, and Joseph Sifakis. Controller synthesis for timed automata. In *Proc. IFAC Symposium on System Structure and Control*, pages 469—474. Elsevier, 1998. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.5874`.

[4] Johan Bengtsson and Wang Yi. Timed automata: Semantics, algorithms and tools. In Jörg Desel, Wolfgang Reisig, and Grzegorz Rozenberg, editors, *Lectures on Concurrency and Petri Nets*, volume 3098 of *Lecture Notes in Computer Science*, pages 87–124. Springer Berlin / Heidelberg, 2004. ISBN 978-3-540-22261-3.

[5] Rameen Beroukhim, Gad Getz, Leia Nghiemphu, Jordi Barretina, Teli Hsueh, David Linhart, Igor Vivanco, Jeffrey C. Lee, Julie H. Huang, Sethu Alexander, Jinyan Du, Tweeny Kau, Roman K. Thomas, Kinjal Shah, Horacio Soto, Sven Perner, John Prensner, Ralph M. Debiasi, Francesca Demichelis, Charlie Hatton, Mark A. Rubin, Levi A. Garraway, Stan F. Nelson, Linda Liau, Paul S. Mischel, Tim F. Cloughesy, Matthew Meyerson, Todd A. Golub, Eric S. Lander, Ingo K. Mellinghoff, and William R. Sellers. Assessing the significance of chromosomal aberrations in cancer: Methodology and application to glioma. *Proceedings of the National Academy of Sciences*, 104(50):20007 –20012, December 2007. doi: 10.1073/pnas.0710052104. URL `http://www.pnas.org/content/104/50/20007.abstract`.

[6] Piergiorgio Bertoli. Planning with extended goals and partial observability. In *In Proceedings of ICAPS04*, pages 270–278, 2004.

[7] Patricia Bouyer. Weighted timed automata: Model-checking and games. *Electronic Notes in Theoretical Computer Science*, 158(0):3–17, 2006. ISSN 1571-0661.

[8] Patricia Bouyer, Deepak D'Souza, P. Madhusudan, and Antoine Petit. Timed control with partial observability. In Warren A. Hunt and Fabio Somenzi, editors, *Computer Aided Verification*, volume 2725, pages 180–192. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-40524-5. URL `http://www.springerlink.com/content/djf805g74a0jwxlq/`.

[9] Franck Cassez, Alexandre David, Kim G. Larsen, Didier Lime, and Jean-François Raskin. Timed control with observation based and stuttering invariant strategies. In Kedar S. Namjoshi, Tomohiro Yoneda, Teruo Higashino, and Yoshio Okamura, editors, *Automated Technology for Verification and Analysis*, volume 4762, pages 192–206. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-75595-1. URL `http://www.springerlink.com/content/b03282876h733023/`.

[10] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Trans. Program. Lang. Syst.*, 8:244–263, April 1986. ISSN 0164-0925.

[11] Edmund Clarke and E. Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. In Dexter Kozen, editor, *Logics of Programs*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71. Springer Berlin / Heidelberg, 1982. ISBN 978-3-540-11212-9.

[12] Edmund M. Clarke, Orna Grumberg, and Doron A. Peled. *Model Checking*. MIT Press, January 1999. ISBN 0262032708.

[13] Elizabeth Comen, Larry Norton, and Joan Massague. Clinical implications of cancer self-seeding. *Nat Rev Clin Oncol*, 8(6):369–377, June 2011. ISSN 1759-4774. doi: 10.1038/nrclinonc.2011.64. URL `http://dx.doi.org/10.1038/nrclinonc.2011.64`.

[14] Karin E. de Visser, Alexandra Eichten, and Lisa M. Coussens. Paradoxical roles of the immune system during cancer development. *Nature Reviews Cancer*, 6(1):24–37, January 2006. ISSN 1474-175X. doi: 10.1038/nrc1782.

[15] Richard Desper, Feng Jiang, Olli-P. Kallioniemi, Holger Moch, Christos H. Papadimitriou, and Alejandro A. Schäffer. Inferring tree models for oncogenesis from comparative genome hybridization data. *Journal of Computational Biology*, 6(1):37–51, January 1999. ISSN 1066-5277, 1557-8666. doi: 10.1089/cmb.1999.6.37. URL `http://www.liebertonline.com/doi/abs/10.1089/cmb.1999.6.37`.

[16] Isaiah J. Fidler. Tumor heterogeneity and the biology of cancer invasion and metastasis. *Cancer Research*, 38(9):2651 –2660, 1978. URL `http://cancerres.aacrjournals.org/content/38/9/2651.abstract`.

[17] Douglas Hanahan and Robert A. Weinberg. The hallmarks of cancer. *Cell*, 100(1):57–70, 2000.

[18] Douglas Hanahan and Robert A. Weinberg. Hallmarks of cancer: The next generation. *Cell*, 144(5):646–674, 2011. ISSN 00928674. doi: 10.1016/j.cell.2011.02.013. URL `http://linkinghub.elsevier.com/retrieve/pii/S0092867411001279`.

[19] Thomas A. Henzinger. The theory of hybrid automata. In *, Eleventh Annual IEEE Symposium on Logic in Computer Science, 1996. LICS '96. Proceedings*, pages 278–292. IEEE, July 1996. ISBN 0-8186-7463-6. doi: 10.1109/LICS.1996.561342.

[20] Thomas A. Henzinger and Peter W. Kopke. Discrete-time control for rectangular hybrid automata. *Theoretical Computer Science*, 221(1-2):369 – 392, 1999. ISSN 0304-3975. doi: 10.1016/S0304-3975(99)00038-9. URL `http://www.sciencedirect.com/science/article/pii/S0304397599000389`.

[21] Thomas A. Henzinger, Peter W. Kopke, Anuj Puri, and Pravin Varaiya. What's decidable about hybrid automata? In *Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, STOC '95, pages 373–382, New York, NY, USA, 1995. ACM. ISBN 0-89791-718-9.

[22] Thomas A. Henzinger, Benjamin Horowitz, and Rupak Majumdar. Rectangular hybrid games. In *In CONCUR 99, LNCS 1664*, pages 320–335. Springer, 1999.

[23] Gloria H. Heppner. Tumor heterogeneity. *Cancer Research*, 44(6):2259–2265, June 1984. URL `http://cancerres.aacrjournals.org/content/44/6/2259.short`.

[24] Gerard Hoffmann and Howard Wong-Toi. Symbolic synthesis of supervisory controllers. In *American Control Conference, 1992*, pages 2789–2793, 1992.

[25] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to automata theory, languages, and computation*. Pearson/Addison Wesley, 2007. ISBN 9780321455369.

[26] Ralph H. Hruban, Michael Goggins, Jennifer Parsons, and Scott E. Kern. Progression model for pancreatic cancer. *Clinical cancer research*, 6(8):2969, 2000.

[27] Iuliana Ionita, Raoul-Sam Daruwala, and Bud Mishra. Mapping Tumor-Suppressor genes with multipoint statistics from Copy-Number–Variation data. *American Journal of Human Genetics*, 79(1):13–22, July 2006. ISSN 0002-9297. PMID: 16773561 PMCID: 1474131.

[28] Shengbing Jiang and Ratnesh Kumar. Supervisory control of discrete event systems with CTL* temporal logic specifications. *SIAM Journal on Control and Optimization*, 44(6):2079–2103, January 2006. doi: 10.1137/S0363012902409982. URL `http://link.aip.org/link/?SJC/44/2079/1`.

[29] Caroline Jose, Nadage Bellance, and Rodrigue Rossignol. Choosing between glycolysis and oxidative phosphorylation: A tumor's dilemma? *Biochimica et Biophysica Acta (BBA) - Bioenergetics*, 1807(6):552 – 561, 2011.

[30] William G. Kaelin Jr. The concept of synthetic lethality in the context of anticancer therapy. *Nat Rev Cancer*, 5(9):689–698, 2005.

[31] Ji Luo, Nicole L. Solimini, and Stephen J. Elledge. Principles of cancer therapy: Oncogene and non-oncogene addiction. *Cell*, 136 (5):823–837, March 2009. ISSN 0092-8674. doi: 10.1016/j.cell.2009.02.024. URL `http://www.sciencedirect.com/science/article/B6WSN-4VS49KS-9/2/2dbe52f01a823f0c41169dbad5978c2f`.

[32] Oded Maler. Control from computer science. *Annual Reviews in Control*, 26(2):175–187, 2002.

[33] George E. Monahan. A survey of partially observable markov decision processes: Theory, models, and algorithms. *Management Science*, 28(1):1–16, 1982. ISSN 00251909.

[34] Nicholas Navin, Jude Kendall, Jennifer Troge, Peter Andrews, Linda Rodgers, Jeanne McIndoo, Kerry Cook, Asya Stepansky, Dan Levy, Diane Esposito, Lakshmi Muthuswamy, Alex Krasnitz, W. Richard McCombie, James Hicks, and Michael Wigler. Tumour evolution inferred by single-cell sequencing. *Nature*, 472(7341):90–94, April 2011. ISSN 0028-0836. doi: 10.1038/nature09807. URL `http://dx.doi.org/10.1038/nature09807`.

[35] R. A. Neher and B. I. Shraiman. Statistical Genetics and Evolution of Quantitative Traits. *ArXiv e-prints*, August 2011.

[36] Larry Norton. Cancer stem cells, Self-Seeding, and decremented exponential growth: Theoretical and clinical implications. *Breast Disease*, 29(-1): 27–36, January 2008. URL `http://iospress.metapress.com/content/T0R3351255U23705`.

[37] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994. ISBN 0471619779.

[38] M.M. Quottrup, T. Bak, and R.I. Zamanabadi. Multi-robot planning : a timed automata approach. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, volume 5, pages 4417–4422, april-1 may 2004.

[39] L. Rabiner and B. Juang. An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4 –16, jan 1986. ISSN 0740-7467.

[40] Atiqur Rahman, Kenneth R. Korzekwa, James Grogan, Frank J. Gonzalez, and James W. Harris. Selective biotransformation of taxol to 6α-hydroxytaxol by human cytochrome p450 2c8. *Cancer research*, 54(21):5543, 1994.

[41] Naren Ramakrishnan, Satish Tadepalli, Layne T. Watson, Richard F. Helm, Marco Antoniotti, and Bud Mishra. Reverse engineering dynamic temporal models of biological processes and their relationships. *Proceedings of the National Academy of Sciences*, 2010. ISSN 0027-8424. doi: 10.1073/pnas.1006283107. URL `http://www.pnas.org/content/early/2010/06/16/1006283107.abstract`.

[42] Richard Simon, Richard Desper, Christos H Papadimitriou, Amy Peng, David S Alberts, Raymond Taetle, Jeffrey M Trent, and Alejandro A Schäffer. Chromosome abnormalities in ovarian adenocarcinoma: III. using breakpoint data to infer and test mathematical models for oncogenesis. *Genes, Chromosomes and Cancer*, 28(1):106–120, May 2000. ISSN 1098-2264. doi: 10.1002/(SICI)1098-2264(200005)28:1⟨106::AID-GCC13⟩3.0.CO;2-S. URL `http://onlinelibrary.wiley.com/doi/10.1002/(SICI)1098-2264(200005)28:1<106::AID-GCC13>3.0.CO;2-S/abstract`.

[43] I. Bernard Weinstein. Addiction to oncogenes–the achilles heal of cancer. *Science*, 297(5578):63 –64, July 2002. doi: 10.1126/science.1073096. URL http://www.sciencemag.org/content/297/5578/63.short.