Beschreibung eines vagen Echtzeit-Hybrid-Systems in TLA$^+$

Lutz Sommerfeld Peter Ladkin

AG Rechnernetze und Verteilte Systeme
Technische Fakultät der Universität Bielefeld
Universitätsstraße 25
33615 Bielefeld

Zusammenfassung

Diese Spezifikation beschreibt ein technischen Gerät, für das keine physikalisch oder mathmatisch genaue Beschreibung vorliegt. Es liegt lediglich eine vage, zum Teil umgangssprachliche Beschreibung der Zusammenhänge vor, da die genauen Prozeßparameter für die Durchführung der Kultivierung unbekannt sind und deren Bestimmung unnötig ist. Darin unterscheidet sich die Aufgabenstellung vom „Steam Boiler Control Problem“. Die vorgestellte Lösung erfolgt auf einer möglichst hohen Abstraktionsebene, so daß die Prozeß-Umwelt-Modellierung als auch die Theorien zur logischen Form von *artifacts* berücksichtigt werden konnten. Als Spezifikationssprache wurde TLA$^+$ gewählt.

1 Problembeschreibung

Der Ausgangspunkt dieser Spezifikation ist die Beschreibung eines Perfusions-Bioreaktors mit einem Fasssungsvermögen von 20 Litern in einer formalen Spezifikationssprache aus der Sicht des benutztenden Biologen. Ein solcher Bioreaktor wird vorzugsweise benutzt, um tierische Körperzellen zu kultivieren. Diese Zellen benötigen für einen schnelles und stetiges Wachstum eine dem natürlichen Lebensraum entsprechende Umgebung. Selbstverständlich hat ein solches Gerät eine ihm zugrunde liegende theoretische Basis (siehe z.B. [Pirt75]), auf der es entwickelt worden ist. Da sich jedoch dieser komplexe biologische Ablauf nur schwer in physikalischen bzw. mathematischen Modell beschreiben läßt, erfolgt die Feineinstellung der einzelnen Komponenten nach den erzielten Kultivierungserfolgen. Der gesamte Aufbau unterliegt daher einem stetigen, wenn auch nur geringen Wandel. So ist z.B. die regelungstechnische Bedeutung der eingestellten PID-Kennwerte für den Zelltechnologen nebensächlich. Er richtet alle Einstellungen danach aus, eine möglichst optimale Vermehrung zu erreichen, ohne sich mit der regelungstechnischen Analyse der Anlage auseinander zu setzen. Der Ansatz dieser Spezifikation ist daher nicht die exakte Beschreibung der technischen Anordnung und deren Einstellungen. Sondern die umgangssprachliche Beschreibung, der für den Zellwachstum wichtigen Bedingungen und deren Realisation mit den vorhandenen technischen Mitteln, aus der Sicht des die Anlage benutzenden Biologen. Diese Informationen wurden in mehreren Experteninterviews ermittelt. Es ist auf diese Weise möglich bei Veränderungen am Aufbau oder der Steuerung des Bioreaktoers festzustellen, ob die von den Biologen gewünschten Kriterien erfüllt werden. Das veränderte System wird nicht auf die Erfüllung eines technischen Modells überprüft, sondern auf die Erfordernisse, die vom Biologen als wichtig erachtet werden. Es wird demnach die rein physikalische Ebene des Geräts verlassen, und auf der abstrakteren Ebene der *artifacts* [Lad96] argumentiert. In diesem Punkt unterscheidet sich diese Spezifikation von den bisherigen Beschreibungen hybrider Systeme, wie z.B. das „Steam boiler control specification problem“[ABL96], welches von Frank Leßke und Stephan Merz in TLA$^+$ realisiert wurde. Dort liegt der Spezifikation ein mathematisch-eindeutig beschriebene Problemstellung zugrunde, welche hier nicht vorhanden ist.

2 Einführung

2.1 Was ist ein Artifact?

In [Sea96] wird beschrieben, daß es zur Beschreibung von *artifacts* nicht ausreichend ist, nur ihre physikalischen Eigenschaften zu berücksichtigen. Ein *artifact* ist demnach ein Gegenstand, welcher neben seiner physikalischen Existenz einen situationsabhängigen Zweck hat. So reicht z.B. die Beschreibung einer mit Tinte gefüllten Miene in einer Plastikhülle, an deren Spitze eine Kugel für die gleichmässige Verteilung der Tinte bei der Bewegung dieser Spitze über einer Oberfläche sorgt, nicht aus, um einen Kugelschreiber zu beschreiben. Erst durch den Zweck des Schreibens, welcher dem Kugelschreiber anhaftet, ist dieser vollständig beschrieben. Zwar gibt es auch Dinge, die nicht zur Klasse der *artifacts* gehören, deren Existenz nicht mit einem eindeutigen Zweck verbunden ist, z.B. der Mensch, bei dem beschriebenen Bioreaktor handelt es sich jedoch unstrittig um ein *artifact*. Daher sollte in der formalen Beschreibung auch die Beschreibung des Zwecks enthalten sein.

2.2 Einführung in TLA+

Die hier benutzte Spezifikationssprache ist TLA$^+$ [Lam95] . Es handelt sich um eine von Lamport entwickelte Sprache, die auf der *Temporal Logic of Actions* TLA[Lam94] basiert. Die Semantik von TLA$^+$ kann mit dem Modell einer unendlichen Folge diskreter, zeitlich linearer Zustände beschrieben werden. Jeder Zustand definiert und unterscheidet sich über die den Zustandsvariablen zugewiesenen Werte.
TLA$^+$ Formeln lassen sich in vier verschiedene Klassen einteilen.

Temporale-Formeln enthalten temporale Operatoren, um über eine unendliche Folge von Zuständen Aussagen zu treffen. Die beiden wichtigsten temporalen Operatoren sind $\Box$ und $\boldsymbol{\exists}$. $\Box A$ besagt daß eine Aussage A für alle folgenden Zustände wahr ist. $\boldsymbol{\exists}$ ist der temporale Existenzquantor (nicht zu verwechseln mit dem logischen $\exists$), welcher es erlaubt Zustandsvariablen innerhalb der Module zu erzeugen (hiding of internal variables). Ein weiterer wichtiger Operatoren ist $\Diamond A \triangleq \neg\Box\neg A$. Er besagt, daß die Aussage A zumindest manchmal in der Zukunft erfüllt sein wird.

Action-Formeln beschreiben den Übergang von einem Zustand in den nächsten.[1] Dem Namen der Variablen, welche den Wert im nächsten Zustand repräsentiert, wird ein *prime* angehängt. Durch $x' = x + 1$ wird beschrieben, daß der Wert von x im folgenden Zustand um 1 erhöht ist. Folgende Bezeichnungen sind zu beachten:

- $[A]_e$ bedeutet, daß entweder A eintritt, oder e ändert seinen Wert nicht.
- $\langle A \rangle_e$ sagt aus, daß ein A Schritt stattfindet und e dabei verändert wird.
- ENABLED A heißt, ein A Schritt ist möglich.
- UNCHANGED e bedeuted, e verändert sich nicht.

[1] Als „nächster Zustand“ wird in TLA$^+$ der nächste Zustand bezeichnet, welcher sich von dem vorhergehenden durch den Inhalt mindestens einer Zustandsvariablen unterscheidet; d.h. TLA$^+$ ist „stuttering invariant“.

Zustands-Formeln sind Formeln welche über einen bestimmten Zustand eine Aussage treffen. Sie enthalten keine *geprimten* Zustandsvariablen, z.B. $y = x + 3$.

Konstante-Formeln enthalten nur Konstanten, also keine Zustandsvariablen.[2]

Desweiteren weist TLA$^+$ einige syntaktische Eigenarten auf, die hier kurz erläutert werden.

Das Modul-Konzept. TLA$^+$ Spezifikationen lassen sich in Module unterteilen, welche sich mit den Befehlen **include** und **import** einbinden lassen. Während **import** einer textuellen Ersetzung gleichkommt, lassen sich mit **include** mehrere Instanzen des selben Moduls unter verschiedenen Namen in einem Modul einbinden. So kann über eine Namenserweiterung (includeName.actionName) auf die in dem „includeten" Modul enthaltenen *actions* zugegriffen werden. Durch zusätzliche Neubenennung der Zustandsvariablen kann zwischen den Zustandsvariablen der verschiedenen Instanzen eindeutig unterschieden werden.

Tupel sind Funktionen. [3] Daher wird beides in TLA$^+$ analog verwendet. So kann $g[i]$ sowohl den Wert der Funktion[4] $g = [x \in S \mapsto e]$ mit Argument i, als auch das i-te Element eines Tupels $g = \langle g_1 \dots g_n \rangle$ bedeuten.

Weiterreichende Informationen über TLA$^+$ finden sich in [Lam94] und [Lam95].

2.3 Der Bioreaktor

Der hier beschriebene membranbegaste 20 Liter Bioreaktor wird über eine analoge elektronische Regelungstechnik gesteuert. Es existiert zusätzlich eine Schnittstelle zu einem PC, welcher mit einer speziellen Regelungssoftware zur Begasungssteuerung ausgestattet ist, die es erlaubt zwischen einer klassischen PID-Regelung und einer benutzerdefinierbaren Fuzzy-Regelung zu wählen. Abgesehen von der Nährlösung, ist die Begasung des Kulturmediums von

[2]Zwar können Konstanten einen beliebigen Wert annehmen, jedoch muß dieser in *allen* Zuständen unverändert bleiben. Hierin unterscheiden sich Konstanten von Zustandsvariablen.

[3]Ein n-Tupel kann als endliche Funktion mit einer n elementigen Definitionsmege aufgefaßt werden, deren n Funktionswerte die Elemente des n-Tupels sind.

[4]Für Werte die nicht in der Definitionsmenge S liegen, liefert die Funktion g einen undefinierten Wert.

größter Bedeutung für das Zellwachstum. Zu diesem Zweck befindet sich im Kulturmedium eine Membran, durch die das Gasgemisch in das flüssige Medium gelangt. Dieses Gas setzt sich aus den vier Komponenten Luft, O_2, N_2 und CO_2 zusammen. Mittels CO_2 wird der pH-Wert des Kulturmediums reguliert. Luft und O_2 dienen dazu, die Luftsättigung (pO_2) im Medium auf einem bestimmten Sollwert zu halten. Da die Tierzellen Sauerstoff verbrauchen und in CO_2 umwandeln, ist es mit steigender Anzahl an Zellen notwendig, den CO_2 Anteil zu veringern und den Luft bzw. O_2 Anteil des Gasgemischs zu erhöhen, um einen Konstanten pH bzw. pO_2 Wert zu erreichen. Damit, trotz schwankender Anteile von Luft, O_2 und CO_2, ein konstanter Gasdurchfluß gehalten werden kann, wird N_2 als *Füllstoff* verwendet, denn es hat keinen Einfluß auf den Kultivierungsprozeß. Zusätzlich zur Membranbegasung gibt es noch die Möglichkeit den Kopfraum oberhalb des Kulturmediums mit Luft oder der Membrangasmischung zu begasen.
Für die Kultivierung ist es weiterhin wichtig, mittels einer Heizung eine bestimmte Temperatur im Medium zu halten. Um eine homogene Verteilung aller Inhaltsstoffe zu gewährleisten, gibt es einen Rotor, welcher das Medium durchmengt. Nähere Einzelheiten zur Kultivierung tierischer Zellen mit diesem Bioreaktor finden sich in [Lüt93].

3 Die Spezifikation

Anhand der Module *Kontrolle* (Abbildungen 1, 2, 3 auf S. 9, 10, 11) und *Kultivierung* (Abbildung 5 auf S. 13) soll exemplarisch die Vorgehensweise bei der Umsetzung des Expertenwissens in eine TLA$^+$ Spezifikation erläutert werden. Zunächst werden jedoch kurz die anderen Module in ihrem Inhalt und ihrer Funktion beschrieben. Auf die Darstellung der eigentlichen TLA$^+$ Spezifikation dieser Module, wurde aus Platzgründen verzichtet. Sie können jedoch unter [SL97] eingesehen werden.

3.1 Modul Echtzeit

Siehe A.1
Für den Kultivierungsprozeß gelten zeitliche Bedingungen, innerhalb derer Veränderungen an den Zustandsvariablen stattfinden. Auch wenn die kritischen Zeitabschnitte nicht im Bereich von Sekundenbruchteilen liegen, sondern vielmehr im Bereich von Minuten bis Stunden, so handelt es sich bei der Kultivierung denoch um einen Echtzeitprozeß. Das Modul *Echtzeit* beschreibt die Behandlung von Echtzeit und *timer* in TLA$^+$, wie sie von Abadi und Lamport in [Lam93] und [AL94] vorgeschlagen wird.

Die meisten Zustandsvariablen sind Größen, die sich nur kontinuierlich ändern können. Andere wiederum, wie z.B. die Anzahl der Zellen, können sich nur diskret verändern. Das Modul beschreibt diese Bedingungen, so daß sie in den anderen Modulen verwendet werden können.

3.2 Modul MiscDef

Siehe A.2
Dieses Modul dient der Auslagerung von zum Teil mathematischen Definitionen. Es wird in verschiedenen Modulen zur Benutzung der Definitonen eingebunden. Das dort eingebundene Modul *Sequences* stammt aus [Lad95].

3.3 Modul Umgebung

Siehe A.3
Der Bioreaktor befindet sich in einer Umgebung, die Einfluß auf den Prozeß der Zellvermehrung haben kann. Damit die Zellkultivierung nicht beeinträchtigt wird, sollte die Lichteinwirkung und die Raumtemperatur innerhalb vorgegebener Grenzen bleiben.

3.4 Modul Medium

Siehe A.4
Ein Bestandteil des Kultivierungsvorganges ist das Kulturmedium. Da es unabhängig von den anderen Teilen betrachtet werden kann, ist es in einem eigenständigen Modul beschrieben. Die in dem Kulturmedium enthaltenen Stoffe sind:

- Tierzellen zur Vermehrung, sowie abgestorbene Zellen.
- Ein Basalmedium, welches ein Gemisch aus vielen für die Kultivierung wichtigen Zutaten ist. Es wird durch eine Funktion dargestellt, welche zu jeder Zutat deren Menge im Basalmedium angibt.
- Bakterien sollten zwar während der gesamten Kultivierung nicht in das Kulturmedium gelangen, sind aber nicht vollständig auszuschliessen.

Die mit dem Kulturmedium verbundenen physikalischen Größen sind:

- Der pH-Wert
- Der pO_2-Wert (Luftsättigung im Kulturmedium)

- Die Temperatur des Kulturmediums

Das Modul *Umgebung* wird eingebunden, da diese auf das Medium einwirkt. Nach der Initialisierung der Zustandsveriablen ist beschrieben, auf welche Art sich diese verändern können.

- Zellen können sich durch Wachstum vermehren
- Zellen können absterben
- Das Basalmedium wird durch die Zellen verbraucht
- Der pH-Wert kann sich verändern
- Der pO_2-Wert kann sich verändern
- Das Kulturmedium kann sich abkühlen

3.5 Modul Bioreaktor

Siehe A.5
Der Bioreaktor weist die zur Einwirkung auf das in ihm enthaltene Kulturmedium nötigen Einrichtungen auf. Er kann, wie auch das Kutlurmedium, als eigenständiger Bestandteil des Gesamtsystems in einem eigenen Modul beschrieben werden. Die den Bioreaktor beschreibenden Zustandsvariablen sind:

- Die Rührerdrehzahl des mit der Begasungsmembran umwickelten Rührers
- Die Gasmischung (Luft,O_2, CO_2, N_2) in der Begasungsmembran und im Kopfraum über dem Kulturmedium
- Der Gasdruck in der Begasungsmembran und im Kopfraum
- Die Flußgeschwindigkeit des Gases in der Begasungsmembran und im Kopfraum
- Das Volumen des Kulturmediums

In dem Bioreaktor befindet sich das Kulturmedium, daher wird das Modul *Medium* eingebunden. Der Initialisierung folgt die Beschreibung der Einwirkungsmöglichkeiten des Bioreaktors auf das Kulturmedium.

- In den Bioreaktor können unabhängig von einander sowohl neues Basalmedium, als auch neue Zellen eingebracht werden

- Eine Entnahme von Kulturmedium ist möglich
- Bei diesen Vorgängen ist eine Verseuchung mit Bakterien nicht auszuschließen
- Das Kulturmedium kann erwärmt werden
- Die Rührerdrehzahl kann verändert werden
- Die Mischung des Gases in der Membran kann verändert werden
- Es kann gewählt werden zwischen der Begasung des Kopfraumes mit Luft oder dem Gasgemisch, welches in der Membran ist
- Die Gasflußgeschwindigkeit in der Membran und im Kopfraum kann einzeln verändert werden
- Der Druck in der Membran und im Kopfraum kann einzeln verändert werden

Es ist beschrieben, daß von den oben aufgezählten Veränderungshandlungen nur das Nachfüllen, die Entnahme und die Verseuchung direkt am Bioreaktor stattfinden können. Alle anderen Veränderungen müssen über die Einstellungen an der Kontrolleinheit erfolgen und werden daher dort ausgeführt.

3.6 Modul Kontrolle

Siehe A.6
In Abbildung 1 auf der nächsten Seite, werden im Abschnitt *parameters* die an der Kontrolle beteiligten Zustandsvariablen und Konstanten definiert. Die mit dem hochgestellten Index s versehenen Variablen stehen für die eingestellten Sollwerte der Zustandsvariablen. Die mit Δ beginnenden Konstanten repräsentieren die maximale Zeitspanne, innerhalb derer vom Sollwert abgewichen werden darf, ohne daß ein negativer Einfluss auf das Wachstum der Zellkultur entsteht. Die übrigen Konstanten sind die Initialisierungswerte für die Sollwerte.
Im Abschnitt *definitions* werden zur besseren Übersicht alle Sollwerte in einem Tuppel *sollVars* zusammengefasst.
Nachfolgend werden die Module *Echtzeit* und *Bioreaktor* eingebunden. Dies ist sinnvoll, da sich die Steuerung auf den Bioreaktor bezieht und zur Einhaltung der oben genannten Zeiten, der im *Echtzeit* Modul definierte *timer*, Abbildung 4 auf Seite 12, benutzt wird. Unter *predicates* werden die Sollwerte mit den dazugehörigen Werten initialisiert. Abbildung 2 auf Seite 10 zeigt die

module *Kontrolle Teil 1*

parameters

$pH, pO_2, temperatur, ruehrerdrehzahl \;:\; Real^{0^+}$ VARIABLE

$membrandruck, membrangasfluss \;:\; Real^{0^+}$ VARIABLE

$kopfraumdruck, kopfraumgasfluss \;:\; Real^{0^+}$ VARIABLE

$pH^s, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s \;:\;$ VARIABLE

$\Delta_{pH}, \Delta_{pO_2}, \Delta_{temp}, \Delta_{drehz}, \Delta_{mDruck}, \Delta_{kDruck}, \Delta_{mFluss}, \Delta_{kFluss} \;:\;$ CONSTANT

$pH_0^s, pO_2{}^s_{(0)}, temp_0^s, drehz_0^s, mDruck_0^s, mFluss_0^s, kDruck_0^s, kFluss_0^s \;:\;$ CONSTANT

definitions

$sollVars \;\triangleq\; \langle pH^s, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s \rangle$

import *Echtzeit*
include *Bioreaktor* **as** *reaktor*

predicates

$$Init \;\triangleq\; \begin{array}{l} \wedge\; pH^s = pH_0^s \\ \vdots \\ \wedge\; kFluss^s = kFluss_0^s \end{array}$$

Tabelle 1: Modul *Kontrolle*, 1. Teil

actions, welche zur Änderung der zur *Kontrolle* gehörenden Zustandsvariablen dienen. Zunächst kann jeder Sollwert auf einen neuen Wert eingestellt werden. Die Wertebereiche sind je nach Sollwert eingeschränkt. Die beiden letzten *actions* tragen dem Umstand Rechnung, daß der pH und der pO_2 Wert nicht direkt, sondern nur indirekt über die Einstellung der Gasmischung beeinflusst werden kann.
Im einzelnen wird beschrieben, daß eine Abweichung vom Sollwert im momentanen Zustand existiert. Das Mambrangas wird mit den entsprechenden Werten eingestellt und die Kopfraumbegasung wird so gewählt, daß im nächsten Zustand der Sollwert erreicht wird. Dies entspricht der Aussage, „Bei nicht eingehaltenem Sollwert, soll durch Einstellung der Gasmischung, der Sollwert wieder ereicht werden.“ Wichtig für die Kultivierung von lebenden Zellen ist es, daß das Kulturmedium von den vorgegebenen Sollwerte nicht länger als einen maximalen Zeitraum abweicht. In Abbildung 3 auf Seite 11 wird nun auf diesen temporalen Aspekt der Steuerung eingegangen. Zunächst wird für alle

module *Kontrolle Teil 2*

actions

$$pH\_soll(s) \triangleq \begin{array}{l} \wedge\ 0 \leq s \leq 14 \\ \wedge\ pH^{s\prime} = s \\ \wedge\ \text{UNCHANGED}\ \langle pO_2, pH, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s \rangle \end{array}$$

$$\vdots$$

$$kFluss\_soll(s) \triangleq \begin{array}{l} \wedge\ 0 \leq s \\ \wedge\ kFluss^{s\prime} = s \\ \wedge\ \text{UNCHANGED}\ \langle pO_2, pH, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s \rangle \end{array}$$

$$pH\_korrektur \triangleq \begin{array}{l} \wedge\ pH \neq pH^s \\ \wedge\ \exists\, co, o, n, l\ :\ reaktor.mGas(co, o, n, l) \\ \wedge\ \exists\, w\ :\ reaktor.kGas(w) \\ \wedge\ pH' = pH^s \\ \wedge\ \text{UNCHANGED}\ \langle pO_2, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s \rangle \end{array}$$

$$pO_2\_korrektur \triangleq \begin{array}{l} \wedge\ pO_2 \neq pO_2^s \\ \wedge\ \exists\, co, o, n, l\ :\ reaktor.mGas(co, o, n, l) \\ \wedge\ \exists\, w\ :\ reaktor.kGas(w) \\ \wedge\ pO_2' = pO_2^s \\ \wedge\ \text{UNCHANGED}\ \langle pH, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s \rangle \end{array}$$

Tabelle 2: Modul *Kontrolle*, 2. Teil

zeitkritischen Zustandsvariablen mit Hilfe eines *timers (VTimer)* beschrieben, daß in allen Zuständen der neu geschaffene *timer* $x = \infty$ ist, wenn die Zustandsvariable dem Sollwert entspricht. Ansonsten ist x gleich dem „Jetztzeitpunkt“ (*now*) plus der maximalen Zeitspanne über die vom Sollwert abgewichen werden darf ($\Delta \ldots$). Der *timer* x erhält einen neuen Wert, wenn der Sollwert erreicht wird. Dadurch, daß dieser *timer* x als ein maximaler *timer* (*MaxTimer*) bezeichnet wird, ist ausgesagt, daß nach Verlassen des Sollwertes innerhalb der Zeit $\Delta \ldots$ der Sollwert wieder erreicht wird. Siehe Abbildung 4 auf Seite 12 zur genauen Definition von *VTimer* und *MaxTimer*. Zur übersichtlicheren Beschreibung der Hauptspezifikation *spec*, werden die *actions* zur Veränderuung der Sollwerte konjugiert. Die Definition *spec* besagt, daß nach der Initialiesierung nur gemäß der oben genannten Definitionen Veränderungen an den Zustandsvariablen geschehen können.

module *Kontrolle Teil 3*

temporal

$$
\begin{array}{lll}
pH\_richtig & \triangleq & \exists\, x : \wedge\ VTimer(x, pH\_korrektur, \Delta_{pH}, pH) \\
 & & \qquad \wedge\ MaxTimer(x) \\
pO_2\_richtig & \triangleq & \exists\, x : \wedge\ VTimer(x, pO_2\_korrektur, \Delta_{pO_2}, pO_2) \\
 & & \qquad \wedge\ MaxTimer(x) \\
temp\_richtig & \triangleq & \exists\, x : \wedge\ VTimer(x, reaktor.heizen(temp^s), \Delta_{temp}, temperatur) \\
 & & \qquad \wedge\ MaxTimer(x) \\
\vdots & & \\
kFluss\_richtig & \triangleq & \exists\, x : \wedge\ VTimer(x, reaktor.kFluss(kFluss^s), \Delta_{kFluss}, kopfraumfluss) \\
 & & \qquad \wedge\ MaxTimer(x) \\
actions & \triangleq & \vee\ \exists\, s : pH\_soll(s) \\
 & & \vdots \\
 & & \vee\ \exists\, s : kFluss\_soll(s) \\
spec & \triangleq & \wedge\ Init \\
 & & \wedge\ pH\_richtig \\
 & & \vdots \\
 & & \wedge\ kFluss\_richtig \\
 & & \wedge\ \Box[actions]_{sollVars}
\end{array}
$$

Tabelle 3: Modul *Kontrolle*, 3. Teil

3.7 Modul Kultivierung

Siehe A.7

Im Abschnitt *parameters* des Moduls *Kultivierung* in Abbildung 5 auf Seite 13, bezeichnet die Konstante $AktiveZellen_{max}$ die Anzahl an Zellen, die durch die Kultivierung erreicht werden sollen, um den Prozeß zu beenden und die Kultur zu ernten. Die Konstante Δ_{doppel} entspricht der Vedoppelungszeit der Tierzellen. Die Variable *letzte* steht für die zuletzt nach einer Verdoppelung erreichte Anzahl an lebenden Zellen im Kultumedium.
Alle an der Kultivierung beteiligten Module werden eingebunden.
Hat sich die Anzahl an lebenden Zellen verdoppelt, so diese der Variablen *letzte* zugewiesen. Somit ist die *verdoppelung* die einzige Möglichkeit, um *letzte* zu verändern. Das *ziel* des *artifact* Bioreaktor ist es, innerhalb der Verdoppelungszeit eine Verdoppelung der Kultur zu erreichen, bis die gewünschte Anzahl an Zellen vorhanden ist. Sollte eine Verunreinigung mit Bakterien stattfinden, so muß der Vorgang abgebrochen und der Bioreaktor entleert

module *Echtzeit (Ausschnitt)*

parameters
now : VARIABLE
∞ : CONSTANT

$$
\begin{array}{lll}
VTimer(x, A, \delta, v) & \triangleq & \land\; x = \textbf{if } \text{ENABLED } \langle A \rangle_v \textbf{ then } now + \delta \\
 & & \qquad\qquad\qquad\qquad\qquad \textbf{else } \infty \\
 & & \land\; \Box[x' = \textbf{if } (\text{ ENABLED } \langle A \rangle_v)' \textbf{ then if } \langle A \rangle_v \textbf{ then } now' + \delta \\
 & & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \textbf{else } x \\
 & & \qquad\qquad\qquad\qquad\qquad\qquad\qquad \textbf{else } \infty\;]_{\langle x, v \rangle} \\
MaxTimer(x) & \triangleq & \Box[(x \neq \infty) \Rightarrow (now' \leq x)]_{now}
\end{array}
$$

Tabelle 4: Modul *Echtzeit* (Ausschnitt), Die *timer*

werden. Entsprechend der zuvor im Modul *Kontrolle* beschriebenen Vorgehensweise, wird mittels der *timer* *VTimer* und *MaxTimer*, welche sich auf die Veränderung der Variablen *letzte* beziehen, ausgedrückt, daß sich in der Zeit Δ_{doppel} die Menge an Zellen verdoppelt. Der Ausdruck

$$
\begin{array}{l}
\Box[(\lor\; aktiveZellen \geq AktiveZellen_{max} \\
\quad \lor\; bakterien \neq 0) \Rightarrow (\land\; \exists\, m, a, t \;:\; bioreaktor.entnahme(a, t, m) \\
\qquad\qquad\qquad\qquad\quad \land\; volumen' = 0)]_{volumen}
\end{array}
$$

beschreibt, daß immer nach erreichen der gewünschten Zellenanzahl oder nach einer Verunreinigung das Kulturmedium restlos entnommen wird.
Unter der Bezeichnung *spec* werden nun die einzlenen Spezifikationen der Module mit dem *ziel* konjugiert und bilden so die Gesamtspezifikation des Kultivierungsprozesses im Bioreaktor.

4 Warum TLA+?

Der entscheidende Vorteil von TLA$^+$ ist, daß es sich um eine Sprache handelt, die auf einer Logik (TLA) basiert. Zu dieser Logik gehören logische Schlußregeln, anhand derer Aussagen miteinander verknüpft und Schlüsse gezogen werden können. Mittels dieser Schlußregeln ist es beispielsweise möglich einen logischen Beweis zu führen, daß eine implementationsnähere Spezifikation eine abstrakte Spezifikation eines Programms implementiert.
Bei der hier vorliegenden Spezifikation handelt es sich nicht um ein Progamm, sondern um ein technisches Gerät. Natürlich könnten auch hier auf un-

module *Kultivierung*

parameters

$bakterien, letzte : Nat^{0^{+}}$ VARIABLE

$volumen : Real^{0^{+}}$ VARIABLE

$AktiveZellen_{max}, \Delta_{doppel}$: CONSTANT

---

import *MiscDef*
import *Echtzeit*
include *Umgebung* **as** *umgebung*
include *Medium* **as** *medium*
include *Bioreaktor* **as** *bioreaktor*
include *Kontrolle* **as** *kontrolle*

---

predicates

$Init \triangleq letzte = 0$

---

actions

$$\begin{aligned} verdoppelung \triangleq\ & \land\ aktiveZellen \geq 2 \cdot letzte \\ & \land\ letzte' = aktiveZellen \\ & \land\ \text{UNCHANGED}\ \langle bakterien, volumen \rangle \end{aligned}$$

---

temporal

$$\begin{aligned} ziel \triangleq\ & \land\ \exists\, x : Vtimer(x, verdoppelung, \Delta_{doppel}, letzte) \\ & \land\ MaxTimer(x) \\ & \land\ \Box[(\lor\ aktiveZellen \geq AktiveZellen_{max} \\ & \qquad \lor\ bakterien \neq 0) \Rightarrow (\land\ \exists\, m, a, t : bioreaktor.entnahme(a, t, m) \\ & \qquad\qquad\qquad\qquad\qquad\quad \land\ volumen' = 0)]_{volumen} \end{aligned}$$

$$\begin{aligned} spec \triangleq\ & \land\ Init \\ & \land\ umgebung.spec \\ & \land\ medium.spec \\ & \land\ bioreaktor.spec \\ & \land\ kontrolle.spec \\ & \land\ ziel \end{aligned}$$

Tabelle 5: Modul *Kultivierung*

terschiedlichen Abstraktionsebenen verschiedene Spezifikationen mittels Implementationsbeweisen verifiziert werden. Zusätzlich bietet sich aber auch die Möglichkeit, einen veränderten Aufbau oder zusätzliche Komponenten des Bioreaktors mit der vorhandenen Spezifikation zu verbinden und ihre Zulässigkeit logischerweise zu beweisen.
Beschreibe das Modul *Erweiterung* die Erweiterung des vorhandenen Aufbaus und sei das Modul *KultivierungNeu* die neue Gesamtspezifikation mit dieser *Erweiterung*, so könnte mittels der Aussage :

$$\begin{aligned}&\exists\, x_1, \cdots, x_n \;:\; \mathit{KultivierungNeu.spec}(x_1, \cdots, x_n, z_1, \cdots z_j)\\ &\quad \Rightarrow \exists\, y_1, \cdots, y_m \;:\; \mathit{Kultivierung.spec}(y_1, \cdots, y_m, F_1(z_1, \cdots z_j), \cdots, F_k(z_1, \cdots z_j))\end{aligned}$$

(wobei die F_i eine bestimmte Funktion, das sogenannte „refinement mapping", ist) bewiesen werden, daß trotz der Erweiterung die veränderte Anlage immer noch den Anforderungen der alten Anlage entspricht. Man erkennt, daß die logische Aussage $A \Rightarrow B$, die Bedeutung hat, A implementiert B. Entsprechend den Anforderungen, unter denen ein solcher Beweis geführt werden soll, ist es auch möglich nur Teile der Gerätespezifikation einem solchen Beweis zu unterziehen. Entscheidend ist, daß durch die logische Beweisführung, die Umsetzung der gewünschten Veränderungen auf ihre Richtigkeit verifiziert werden kann.

5 Zusammenfassung und Ausblick

Trotz der nur vagen Beschreibung der unterschiedlichen Komponenten des Bioreaktors, ist es gelungen eine inhaltlich zutreffende und formal korrekte[5] TLA$^+$ Spezifikation zu schreiben. Dabei konnte auf dem Abstraktionsniveau des den Reaktor benutzenden Biologen verblieben werden. Alle technischen Komponenten werden nur so detailliert beschreibenden, wie es erforderlich ist, um auf einer Abstraktionsstufe mit dem Nutzen des Bioreaktors zu stehen. Hierdurch ist es möglich, auch der aktuellen philosophischen Grundlagenforschung zum Thema formale Beschreibungen, welche unter anderem von Searle [Sea96] und Ladkin [Lad96] in der Behandlung der *artifacts* stattfindet, gerecht zu werden. Es wurde sowohl das Zusammenspiel der Umwelt mit dem Kulturmedium, als auch die den Prozeß bestimmenden Echtzeitbedingungen berücksichtitg.
Auf diesem Ergebnis aufbauend, ist es nun möglich Veränderungen, welche am Bioreaktor oder einer seiner Komponenten vorgenommen werden, mittels

[5] „formal korrekt" bedeutet hier, daß ausschliesslich TLA$^+$ Sprachelemente verwendet wurden und somit eine klare Zuordnung von Syntax und Semantik für die gesamte Spezifikation sichergestellt ist.

Korrektheisbeweise zu überprüfen. Dabei kann auf Grund der Modulsruktur nur ein begrenzter Teil als auch die gesamte Anlage in die Überprüfung mit einbezogen werden. Neu ist dabei, daß nicht die technischen Anordnungen miteinander verglichen werden, sondern daß die Veränderungen mit den im Modul *Kultivierung* durch *spec* beschriebenen Anforderungen übereinstimmen. Dies gewährleistet, daß der veränderte Aufbau weiterhin dem *artifact* Bioreaktor entspricht, und somit auf einer abstrakten Bedeutungsebene keine ungewollte Veränderung stattgefunden hat. Solche Beweise sind zur Zeit zwar noch sehr langwierig und arbeitsintensiv [Lad95], könnten jedoch durch eine entsprechende Untersützung von automatischen Beweisern und „proofchecker" vereinfacht werden.

Literatur

[ABL96] Jean-Raymond Abrial, Egon Börger, Hans Langmaack: editors. Formal Methods for Industrial Applications – Specifying and Programming the Steam Boiler Control. Lecture Notes in Computer Science, Vol. 1165, Springer Verlag, 1996.

[AL94] Martín Abadi, Leslie Lamport. An Old-Fashioned Recipe for Real Time. ACM Transactions on Programming Languages and Systems, 16(5):1543-1571, September 1994.

[Lad95] Peter Ladkin. Formal But Lively Buffers in TLA$^+$. Technical Report 96-07 der Technischen Fakultät, Arbeitsgruppe RVS, Universität Bielefeld.
Available from `http://www.rvs.uni-bielefeld.de`.

[Lad96] Peter Ladkin. The Success and Failure of Artifacts. Technical Report RVS-Occ-97-04 der Technischen Fakultät, Arbeitsgruppe RVS, Universität Bielefeld. Available from `http://www.rvs.uni-bielefeld.de`.

[Lam93] Leslie Lamport. Hybrid Systems in TLA$^+$. Grossmann, Nerode, Ravn, Rischl: editors. Hybrid Systems, Lecture Notes in Computer Science, Vol. 736. Springer Verlag, 1993.

[Lam94] Leslie Lamport. The temporal logic of actions. ACM Transactions on Programming Languages and Systems, 16(3):872-923, Mai 1994.

[Lam95] Leslie Lamport. TLA$^+$.
Available from `http://www.research.digital.com/SRC/tla/`.

[Lüt93] Dirk Lütkemeyer. Mikroporöse Membranionenaustauscher zur schnellen Reinigung von Proteinen aus tierischen Zellkulturen. Dissertation an der Technischen Fakultät der Univeristät Bielefeld, Juli 1993.

[Pirt75] S. John Pirt. Principles of Microbe and Cell Cultivation. Oxford: Blackwell Scientific Publications, 1975.

[Sea96] John R. Searle. The Construction of Social Reality. London: Penguin Books, 1996.

[SL97] Lutz Sommerfeld, Peter Ladkin.
Beschreibung eines vagen Echtzeit-Hybrid-Systems in TLA$^+$.
Available from `http://www.rvs.uni-bielefeld.de/~lutz/`.

A Die Module

Bei den Modulen in diesem Anhang handelt es sich um vorläufige Ergebnisse. Kleinere Veränderungen sind daher nicht auszuschließen.

A.1 Echtzeit

module *Echtzeit*

parameters

now : VARIABLE

∞ : CONSTANT

import *MiscDef*

assumption

$$InfinityUnReal \triangleq \infty \notin Real$$

actions

$$\begin{aligned}
RT(v) \triangleq\ & \land\ now \in Real \\
& \land\ \Box[\land\ now' \in \{r \in Real : r > now\} \\
& \qquad \land\ v' = v]_{now} \\
& \land\ \forall\, t \in Real : \Diamond(now > t)
\end{aligned}$$

$$\begin{aligned}
VTimer(x, A, \delta, v) \triangleq\ & \land\ x = \textbf{if } \text{ENABLED } \langle A \rangle_v \textbf{ then } now + \delta \\
& \qquad\qquad \textbf{else } \infty \\
& \land\ \Box[x' = \textbf{if } (\text{ ENABLED } \langle A \rangle_v)' \textbf{ then if } \langle A \rangle_v \textbf{ then } now' + \delta \\
& \qquad\qquad\qquad\qquad\qquad\qquad \textbf{else } x \\
& \qquad\qquad\qquad\qquad \textbf{else } \infty\ ]_{\langle x, v \rangle}
\end{aligned}$$

$$MaxTimer(x) \triangleq \Box[(x \neq \infty) \Rightarrow (now' \leq x)]_{now}$$

$$\begin{aligned}
kontinuierlich(v) \triangleq\ & \forall\, i \in 1..Len(v) : \\
& \Box[\forall\, \delta \in Real^{+} : \exists\, \epsilon \in Real : now' - now < \epsilon \Rightarrow |v[i]' - v[i]| < \delta]_{v[i]}
\end{aligned}$$

A.2 MiscDef

module *MiscDef*

import *Reals*
import *Naturals*
import *Sequences*[6]

definitions

[6] from Peter Ladkin

$Real^{+} \triangleq \{x \in Real : x > 0\}$
$Real^{0^{+}} \triangleq \{x \in Real : x \geq 0\}$
$Nat^{+} \triangleq \{n \in Nat : n > 0\}$
$Nat^{0^{+}} \triangleq \{n \in Nat : n \geq 0\}$
$|z| \triangleq \textbf{if } z < 0 \textbf{ then } -z$
$\qquad\qquad \textbf{else } z$

A.3 Umgebung

module *Umgebung*

parameters
$temperatur, helligkeit$: VARIABLE
$Temperatur_0, Helligkeit_0, MaxHelligkeit$: CONSTANT

predicates
$Init \triangleq \wedge\ temperatur = Temperatur_0$
$\qquad\quad \wedge\ helligkeit = Helligkeit_0$

definitions
$tempBed \triangleq 10°C \leq temperatur' \leq 35°C$
$lichtBed \triangleq helligkeit' \leq MaxHelligkeit$

temporal
$vars \triangleq \langle temperatur, helligkeit \rangle$
$spec \triangleq \wedge\ Init$
$\qquad\quad \wedge\ \Box[tempBed \wedge lichtBed]_{vars}$
$\qquad\quad \wedge\ RT(vars)$
$\qquad\quad \wedge\ kontinuierlich(vars)$

A.4 Medium

module *Medium*

parameters
$aktiveZellen, toteZellen, bakterien : Nat^{0^{+}}$ VARIABLE
$pH, pO_2, temperatur, raumtemperatur : Real^{+}$ VARIABLE
$basalmedium : Zutaten \cup Gifte \rightarrow Real^{0^{+}}$ VARIABLE
$Zutaten, Gifte, Basalmedium_0, pH_0, pO_{2(0)}, temperatur_0$: CONSTANT
$AktiveZellen_0, ToteZellen_0$: CONSTANT

import *Echtzeit*
include *Umgebung* **as** *umwelt* **with** *raumtemperatur* $\leftarrow$ *temperatur*

---

predicates

$$
\begin{array}{lll}
Init & \triangleq & \land\ aktiveZellen = AktiveZellen_0 \\
 & & \land\ toteZellen = ToteZellen_0 \\
 & & \land\ basalmedium = Basalmedium_0 \\
 & & \land\ pH = pH_0 \\
 & & \land\ pO_2 = pO_{2(0)} \\
 & & \land\ temperatur = temperatur_0 \\
 & & \land\ bakterien = 0
\end{array}
$$

---

actions

$$
\begin{array}{lll}
wachstum & \triangleq & \land\ aktiveZellen' = aktiveZellen + 1 \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, toteZellen, basalmedium, pH, pO_2, \\
 & & \qquad temperatur, bakterien \rangle \\
sterben & \triangleq & \land\ aktiveZellen > 0 \\
 & & \land\ aktiveZellen' = aktiveZellen - 1 \\
 & & \land\ toteZellen' = toteZellen + 1 \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, basalmedium, pH, pO_2, temperatur, bakterien \rangle \\
verbrauch & \triangleq & \land\ \exists z \in Zutaten\ :\ 0 \leq basalmedium'[z] < basalmedium[z] \\
 & & \land\ \forall x \in Zutaten \cup Gifte\ :\ (x = z \lor \text{UNCHANGED}\ basalmedium[x]) \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, aktiveZellen, toteZellen, \\
 & & \qquad pH, pO_2, temperatur, bakterien \rangle \\
ausscheidung & \triangleq & \land\ \exists g \in Gifte\ :\ basalmedium'[g] > basalmedium[g] \\
 & & \land\ \forall x \in Zutaten \cup Gifte\ :\ (x = g \lor \text{UNCHANGED}\ basalmedium[x]) \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, aktiveZellen, toteZellen, \\
 & & \qquad pH, pO_2, temperatur, bakterien \rangle \\
pHaenderung & \triangleq & \land\ 0 \leq pH' \leq 14 \\
 & & \land\ pH' \neq pH \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, aktiveZellen, toteZellen, basalmedium, \\
 & & \qquad pO_2, temperatur, bakterien \rangle \\
pO_2aenderung & \triangleq & \land\ 0 \leq pO_2' \leq 500 \\
 & & \land\ pO_2' \neq pO_2 \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, aktiveZellen, toteZellen, basalmedium, \\
 & & \qquad pH, temperatur, bakterien \rangle \\
abkuehlung & \triangleq & \land\ raumtemperatur \leq temperatur' < temperatur \\
 & & \land\ \text{UNCHANGED}\ \langle raumtemperatur, aktiveZellen, toteZellen, \\
 & & \qquad basalmedium, pH, pO_2, bakterien \rangle
\end{array}
$$

---

temporal

$$
\begin{array}{lll}
vars1 & \triangleq & \langle pH, pO_2, temperatur, raumtemperatur \rangle \\
vars & \triangleq & vars1 \circ \langle basalmedium, aktiveZellen, toteZellen, bakterien \rangle
\end{array}
$$

$$
\begin{aligned}
actions \;\triangleq\;& \lor\ wachstum \\
& \lor\ sterben \\
& \lor\ verbrauch \\
& \lor\ pHaenderung \\
& \lor\ pO_2aenderung \\
& \lor\ abkuehlung
\end{aligned}
$$

$$
\begin{aligned}
spec \;\triangleq\;& \land\ Init \\
& \land\ \Box[actions]_{vars} \\
& \land\ RT(vars) \\
& \land\ kontinuierlich(vars1)
\end{aligned}
$$

A.5 Bioreaktor

module *Bioreaktor*

parameters

$ruehrerdrehzahl, membrandruck, membrangas, membrangasfluss\ :\ Real^{0^+}$ VARIABLE

$temperatur, kopfraumdruck, kopfraumgas, kopfraumgasfluss, volumen\ :\ Real^{0^+}$ VARIABLE

$aktiveZellen, toteZellen, bakterien\ :\ Nat^{0^+}$ VARIABLE

$basalmedium\ :\ Zutaten \rightarrow Real^{0^+}$ VARIABLE

$Zutaten, Volumen_0, Drehzahl_0\ :$ CONSTANT

$Membrandruck_0, Membrangas_0, Membrangasfluss_0\ :$ CONSTANT

$Kopfraumdruck_0, Kopfraumgas_0, Kopfraumgasfluss_0\ :$ CONSTANT

import *Echtzeit*
include *Medium* **as** *medium*

definitions

$Maxvolumen \;\triangleq\; 20$

predicates

$$
\begin{aligned}
Init \;\triangleq\;& \land\ ruehrerdrehzahl = Drehzahl_0 \\
& \land\ volumen = Volumen_0 \\
& \land\ membrandruck = Membrandruck_0 \\
& \land\ membrangas = Membrangas_0 \\
& \land\ membrangasfluss = Membrangasfluss_0 \\
& \land\ kopfraumdruck = Kopfraumdruck_0 \\
& \land\ kopfraumgas = Kopfraumgas_0 \\
& \land\ kopfraumgasfluss = Kopfraumgasfluss_0
\end{aligned}
$$

actions

$$
\begin{array}{lll}
\mathit{neueZellen}(a, t) & \triangleq & \land\ a \in \mathit{Nat}^{+} \\
 & & \land\ t \in \mathit{Nat}^{0^{+}} \\
 & & \land\ \mathit{aktiveZellen}' = \mathit{aktiveZellen} + a \\
 & & \land\ \mathit{toteZellen}' = \mathit{toteZellen} + t \\
 & & \land\ \mathit{volumen} < \mathit{volumen}' \leq \mathit{Maxvolumen} \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{ruehrerdrehzahl}, \mathit{kopfraumdruck}, \mathit{membrandruck}, \\
 & & \qquad \mathit{membrangas}, \mathit{membrangasfluss}, \mathit{kopfraumgas}, \\
 & & \qquad \mathit{kopfraumgasfluss}, \mathit{basalmedium}, \mathit{temperatur}, \mathit{bakterien} \rangle
\end{array}
$$

$$
\begin{array}{lll}
\mathit{neuesMedium}(m) & \triangleq & \land\ m \in \mathit{Zutaten} \rightarrow \mathit{Real}^{0^{+}} \\
 & & \land\ \forall\, x \in \mathit{Zutaten} : \mathit{basalmedium}[x]' = \mathit{basalmedium}[x] + m[x] \\
 & & \land\ \mathit{volumen} < \mathit{volumen}' \leq \mathit{Maxvolumen} \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{ruehrerdrehzahl}, \mathit{kopfraumdruck}, \mathit{membrandruck}, \\
 & & \qquad \mathit{membrangas}, \mathit{membrangasfluss}, \mathit{kopfraumgas}, \\
 & & \qquad \mathit{kopfraumgasfluss}, \mathit{aktiveZellen}, \mathit{toteZellen}, \\
 & & \qquad \mathit{temperatur}, \mathit{bakterien} \rangle
\end{array}
$$

$$
\begin{array}{lll}
\mathit{entnahme}(a, t, m) & \triangleq & \land\ t, a \in \mathit{Nat}^{0^{+}} \\
 & & \land\ a \leq \mathit{aktiveZellen} \\
 & & \land\ t \leq \mathit{toteZellen} \\
 & & \land\ m \in \mathit{Zutaten} \rightarrow \mathit{Real}^{+} \\
 & & \land\ \mathit{aktiveZellen}' = \mathit{aktiveZellen} - a \\
 & & \land\ \mathit{toteZellen}' = \mathit{toteZellen} - t \\
 & & \land\ \forall\, x \in \mathit{Zutaten} : \land\ \mathit{basalmedium}[x]' = \mathit{basalmedium}[x] - m[x] \\
 & & \qquad\qquad\qquad\quad \land\ 0 \leq \mathit{basalmedium}[x]' \\
 & & \land\ 0 \leq \mathit{volumen}' < \mathit{volumen} \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{ruehrerdrehzahl}, \mathit{kopfraumdruck}, \mathit{membrandruck}, \\
 & & \qquad \mathit{membrangas}, \mathit{membrangasfluss}, \mathit{kopfraumgas}, \\
 & & \qquad \mathit{kopfraumgasfluss}, \mathit{temperatur}, \mathit{bakterien} \rangle
\end{array}
$$

$$
\begin{array}{lll}
\mathit{verseuchung}(n) & \triangleq & \land\ n \in \mathit{Nat}^{+} \\
 & & \land\ \mathit{bakterien}' = \mathit{bakterien} + n \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{ruehrerdrehzahl}, \mathit{kopfraumdruck}, \mathit{membrandruck}, \\
 & & \qquad \mathit{membrangas}, \mathit{membrangasfluss}, \mathit{kopfraumgas}, \\
 & & \qquad \mathit{kopfraumgasfluss}, \mathit{aktiveZellen}, \mathit{toteZellen}, \\
 & & \qquad \mathit{basalmedium}, \mathit{temperatur}, \mathit{volumen} \rangle
\end{array}
$$

$$
\begin{array}{lll}
\mathit{heizen}(t) & \triangleq & \land\ t > \mathit{temperatur} \\
 & & \land\ \mathit{temperatur}' = t \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{ruehrerdrehzahl}, \mathit{kopfraumdruck}, \mathit{membrandruck}, \\
 & & \qquad \mathit{membrangas}, \mathit{membrangasfluss}, \mathit{kopfraumgas}, \\
 & & \qquad \mathit{kopfraumgasfluss}, \mathit{aktiveZellen}, \mathit{toteZellen}, \\
 & & \qquad \mathit{basalmedium}, \mathit{bakterien}, \mathit{volumen} \rangle
\end{array}
$$

$$
\begin{array}{lll}
\mathit{drehzahl}(d) & \triangleq & \land\ d \in \mathit{Real}^{0^{+}} \\
 & & \land\ d \neq \mathit{ruehrerdrehzahl} \\
 & & \land\ \mathit{ruehrerdrehzahl}' = d \\
 & & \land\ \text{UNCHANGED}\ \langle \mathit{kopfraumdruck}, \mathit{membrandruck}, \mathit{membrangas}, \\
 & & \qquad \mathit{membrangasfluss}, \mathit{kopfraumgas}, \mathit{kopfraumgasfluss}, \\
 & & \qquad \mathit{aktiveZellen}, \mathit{toteZellen}, \mathit{basalmedium}, \\
 & & \qquad \mathit{temperatur}, \mathit{bakterien}, \mathit{volumen} \rangle
\end{array}
$$

$$
\begin{array}{lll}
mGas(co, o, n, l) & \triangleq & \land\ co, o, n, l \in Real^{0^+} \\
 & & \land\ co + o + n + l = 100 \\
 & & \land\ \langle co, o, n, l\rangle \neq membrangas \\
 & & \land\ membrangas' = \langle co, o, n, l\rangle \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, kopfraumdruck, membrandruck, \\
 & & \qquad membrangasfluss, kopfraumgasfluss, aktiveZellen, \\
 & & \qquad toteZellen, basalmedium, temperatur, \\
 & & \qquad bakterien, volumen, kopfraumgas\rangle
\end{array}
$$

$$
\begin{array}{lll}
kGas(wahl) & \triangleq & \land\ wahl \in \{\text{“luft”}, \text{“gemisch”}\} \\
 & & \land\ kopfraumgas' = \textbf{if}\ wahl = \text{“luft”}\ \textbf{then}\ \langle 0, 0, 0, 100\rangle \\
 & & \qquad\qquad\qquad\qquad\qquad\qquad \textbf{else}\ \ membrangas \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, kopfraumdruck, membrandruck, \\
 & & \qquad membrangasfluss, kopfraumgasfluss, aktiveZellen, \\
 & & \qquad toteZellen, basalmedium, temperatur, \\
 & & \qquad bakterien, volumen, membrangas\rangle
\end{array}
$$

$$
\begin{array}{lll}
mFluss(m) & \triangleq & \land\ m \in Real^{0^+} \\
 & & \land\ m \neq membrangasfluss \\
 & & \land\ membrangasfluss' = m \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, kopfraumdruck, membrandruck, \\
 & & \qquad membrangas, kopfraumgas, aktiveZellen, toteZellen, \\
 & & \qquad basalmedium, temperatur, bakterien, volumen, \\
 & & \qquad kopfraumgasfluss\rangle
\end{array}
$$

$$
\begin{array}{lll}
kFluss(m) & \triangleq & \land\ m \in Real^{0^+} \\
 & & \land\ m \neq kopfraumgasfluss \\
 & & \land\ kopfraumgasfluss' = m \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, kopfraumdruck, membrandruck, \\
 & & \qquad membrangas, kopfraumgas, aktiveZellen, toteZellen, \\
 & & \qquad basalmedium, temperatur, bakterien, volumen, \\
 & & \qquad membrangasfluss\rangle
\end{array}
$$

$$
\begin{array}{lll}
mDruck(d) & \triangleq & \land\ d \in Real^{+} \\
 & & \land\ d \neq membrandruck \\
 & & \land\ membrandruck' = d \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, membrangas, membrangasfluss, \\
 & & \qquad kopfraumgas, kopfraumgasfluss, kopfraumdruck, \\
 & & \qquad aktiveZellen, toteZellen, basalmedium, \\
 & & \qquad temperatur, bakterien, volumen\rangle
\end{array}
$$

$$
\begin{array}{lll}
kDruck(d) & \triangleq & \land\ d \in Real^{+} \\
 & & \land\ d \neq kopfraumdruck \\
 & & \land\ kopfraumdruck' = d \\
 & & \land\ \text{UNCHANGED}\ \langle ruehrerdrehzahl, membrangas, membrangasfluss, \\
 & & \qquad kopfraumgas, kopfraumgasfluss, membrandruck \\
 & & \qquad aktiveZellen, toteZellen, basalmedium, \\
 & & \qquad temperatur, bakterien, volumen\rangle
\end{array}
$$

---

temporal

$$
\begin{array}{lll}
vars1 & \triangleq & \langle ruehrerdrehzahl, kopfraumdruck, membrandruck, membrangas, membrangasfluss, \\
 & & \ kopfraumgas, kopfraumgasfluss, temperatur, volumen\rangle
\end{array}
$$

$$\begin{aligned}
vars &\triangleq vars1 \circ \langle basalmedium, aktiveZellen, toteZellen, bakterien\rangle \\
actions &\triangleq \lor \exists\, a, t : neueZellen(a, t) \\
&\quad \lor \exists\, m : neuesMedium(m) \\
&\quad \lor \exists\, a, t, m : entnahme(a, t, m) \\
&\quad \lor \exists\, n : verseuchung(n) \\
spec &\triangleq \land\ Init \\
&\quad \land\ \Box[actions]_{vars} \\
&\quad \land\ RT(vars) \\
&\quad \land\ kontinuierlich(vars1)
\end{aligned}$$

A.6 Kontrolle

module *Kontrolle*

parameters

$pH, pO_2, temperatur, ruehrerdrehzahl : Real^{0^+}$ VARIABLE
$membrandruck, membrangasfluss : Real^{0^+}$ VARIABLE
$kopfraumdruck, kopfraumgasfluss : Real^{0^+}$ VARIABLE
$pH^s, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s$: VARIABLE
$\Delta_{pH}, \Delta_{pO_2}, \Delta_{temp}, \Delta_{drehz}, \Delta_{mDruck}, \Delta_{kDruck}, \Delta_{mFluss}, \Delta_{kFluss}$: CONSTANT
$pH_0^s, pO_{2(0)}^s, temp_0^s, drehz_0^s, mDruck_0^s, mFluss_0^s, kDruck_0^s, kFluss_0^s$: CONSTANT

definitions

$$sollVars \triangleq \langle pH^s, pO_2^s, temp^s, drehz^s, mDruck^s, mFluss^s, kDruck^s, kFluss^s\rangle$$

import *Echtzeit*
include *Bioreaktor* **as** *reaktor*

predicates

$$\begin{aligned}
Init \triangleq\ &\land\ pH^s = pH_0^s \\
&\land\ pO_2^s = pO_{2(0)}^s \\
&\land\ temp^s = temp_0^s \\
&\land\ drehz^s = drehz_0^s \\
&\land\ mDruck^s = mDruck_0^s \\
&\land\ mFluss^s = mFluss_0^s \\
&\land\ kDruck^s = kDruck_0^s \\
&\land\ kFluss^s = kFluss_0^s
\end{aligned}$$

actions

$$\begin{aligned}
pH_soll(s) &\triangleq \land\ 0 \le s \le 14 \\
&\quad\ \land\ pH^{s\prime} = s \\
pO_2_soll(s) &\triangleq \land\ 0 \le s \le 100 \\
&\quad\ \land\ pO_2^{s\prime} = s
\end{aligned}$$

$$
\begin{aligned}
temp\_soll(s) &\triangleq \land\ 10 \le s \le 80 \\
&\quad\ \land\ temp^{s\prime} = s \\
drehz\_soll(s) &\triangleq \land\ 0 \le s \le 50 \\
&\quad\ \land\ drehz^{s\prime} = s \\
mDruck\_soll(s) &\triangleq \land\ 0 \le s \\
&\quad\ \land\ mDruck^{s\prime} = s \\
kDruck\_soll(s) &\triangleq \land\ 0 \le s \\
&\quad\ \land\ kDruckd^{s\prime} = s \\
mFluss\_soll(s) &\triangleq \land\ 0 \le s \\
&\quad\ \land\ mFluss^{s\prime} = s \\
kFluss\_soll(s) &\triangleq \land\ 0 \le s \\
&\quad\ \land\ kFluss^{s\prime} = s \\
pH\_korrektur &\triangleq \land\ pH \neq pH^{s} \\
&\quad\ \land\ \exists\, co, o, n, l \,:\, reaktor.mGas(co, o, n, l) \\
&\quad\ \land\ \exists\, w \,:\, reaktor.kGas(w) \\
&\quad\ \land\ pH' = pH^{s} \\
&\quad\ \land\ \text{UNCHANGED}\ pO_2 \\
pO_2\_korrektur &\triangleq \land\ pO_2 \neq pO_2^{s} \\
&\quad\ \land\ \exists\, co, o, n, l \,:\, reaktor.mGas(co, o, n, l) \\
&\quad\ \land\ \exists\, w \,:\, reaktor.kGas(w) \\
&\quad\ \land\ pO_2' = pO_2^{s} \\
&\quad\ \land\ \text{UNCHANGED}\ pH
\end{aligned}
$$

---

temporal

$$
\begin{aligned}
pH\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, pH\_korrektur, \Delta_{pH}, pH) \\
&\qquad\qquad \land\ MaxTimer(x) \\
pO_2\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, pO_2\_korrektur, \Delta_{pO_2}, pO_2) \\
&\qquad\qquad \land\ MaxTimer(x) \\
temp\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.heizen(temp^{s}), \Delta_{temp}, temperatur) \\
&\qquad\qquad \land\ MaxTimer(x) \\
drehz\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.drehzahl(drehz^{s}), \Delta_{drehz}, ruehrerdrehzahl) \\
&\qquad\qquad \land\ MaxTimer(x) \\
mDruck\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.mDruck(mDruck^{s}), \Delta_{mDruck}, membrandruck) \\
&\qquad\qquad \land\ MaxTimer(x) \\
kDruck\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.kDruck(kDruckd^{s}), \Delta_{kDruck}, kopfraumdruck) \\
&\qquad\qquad \land\ MaxTimer(x) \\
mFluss\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.mFluss(mFluss^{s}), \Delta_{mFluss}, membranfluss) \\
&\qquad\qquad \land\ MaxTimer(x) \\
kFluss\_richtig &\triangleq \exists\, x \,:\, \land\ VTimer(x, reaktor.kFluss(kFluss^{s}), \Delta_{kFluss}, kopfraumfluss) \\
&\qquad\qquad \land\ MaxTimer(x)
\end{aligned}
$$

$$actions \triangleq \begin{array}{l} \lor \exists s : pH\_soll(s) \\ \lor \exists s : pO_2\_soll(s) \\ \lor \exists s : temp\_soll(s) \\ \lor \exists s : drehz\_soll(s) \\ \lor \exists s : mDruck\_soll(s) \\ \lor \exists s : kDruck\_soll(s) \\ \lor \exists s : mFluss\_soll(s) \\ \lor \exists s : kFluss\_soll(s) \end{array}$$

$$spec \triangleq \begin{array}{l} \land Init \\ \land pH\_richtig \\ \land pO_2\_richtig \\ \land temp\_richtig \\ \land drehz\_richtig \\ \land mDruck\_richtig \\ \land kDruck\_richtig \\ \land mFluss\_richtig \\ \land kFluss\_richtig \\ \land \Box[actions]_{sollVars} \end{array}$$

A.7 Kultivierung

module *Kultivierung*

parameters

$bakterien, letzte : Nat^{0^+}$ VARIABLE

$volumen : Real^{0^+}$ VARIABLE

$AktiveZellen_{max}, \Delta_{doppel}$: CONSTANT

import *MiscDef*
import *Echtzeit*
include *Umgebung* **as** *umgebung*
include *Medium* **as** *medium*
include *Bioreaktor* **as** *bioreaktor*
include *Kontrolle* **as** *kontrolle*

predicates

$Init \triangleq letzte = 0$

actions

$$verdoppelung \triangleq \begin{array}{l} \land aktiveZellen \geq 2 \cdot letzte \\ \land letzte' = aktiveZellen \\ \land \text{UNCHANGED } \langle bakterien, volumen \rangle \end{array}$$

temporal

$$
\begin{array}{lll}
ziel & \triangleq & \land\ \exists\, x\ :\ Vtimer(x, verdoppelung, \Delta_{doppel}, letzte) \\
& & \land\ MaxTimer(x) \\
& & \land\ \Box[(\lor\ aktiveZellen \geq AktiveZellen_{max} \\
& & \qquad \lor\ bakterien \neq 0) \Rightarrow (\land\ \exists\, m, a, t\ :\ bioreaktor.entnahme(a, t, m) \\
& & \qquad\qquad\qquad\qquad\qquad\quad \land\ volumen' = 0)]_{volumen} \\
spec & \triangleq & \land\ Init \\
& & \land\ umgebung.spec \\
& & \land\ medium.spec \\
& & \land\ bioreaktor.spec \\
& & \land\ kontrolle.spec \\
& & \land\ ziel
\end{array}
$$