Формальный дедуктивный анализ автоматного алгоритма управления генератором эндогаза с помощью платформы Rodin

Часть 3. Построение формальной теории для алгоритма управления

Максим Нейзов (neyzov.max@gmail.com)

Формальный дедуктивный анализ представляет собой строгий математический подход к верификации алгоритмов: алгоритм описывается с помощью аксиом, а требуемые свойства доказываются как теоремы. Цель представленного анализа – доказать соответствие алгоритма управления предъявляемым требованиям надежности и безопасности. В статье выполнено построение формальной теории для алгоритма управления: проведена аксиоматизация алгоритма, формализованы требования, продемонстрировано доказательство теоремы с помощью платформы Rodin.

Введение

В предыдущих частях статьи были определены требования надёжности и безопасности [1], предъявляемые к алгоритму, рассмотрены автоматный алгоритм управления и платформа Rodin [2]. В настоящей работе для верификации алгоритма выполняется формальный дедуктивный анализ, который состоит в построении формальной аксиоматической теории. Для этого алгоритм описывается с помощью аксиом, выполняется формализация требований и проводится автоматизированное доказательство теорем с помощью платформы Rodin.

Аксиоматика автоматов

Автоматы A и B [2] задаются с помощью аксиом в контексте (см. листинг 1 и 2 соответственно). Назначение аксиом представлено в таблице 1. Если аксиоматика противоречива, то модель не может быть запущена на исполнение.

Листинг 1

```
axm1: partition(A, {A1},{A2},{A3},{A4})
axm2: InA = (BOOL × BOOL × BOOL)
axm3: a_1_2 = {x1  ↦ x2 ↦ x3 | x1=TRUE ∧ x2=TRUE ∧ x3=TRUE}
axm4: a_2_3 = {x1 ↦ x2 ↦ x3 | x1=TRUE ∧ x2=TRUE ∧ x3=FALSE}
axm5: a_2_4 = {x1 ↦ x2 ↦ x3 | x1=TRUE ∧ x2=FALSE ∧ x3=TRUE}
axm6: a_234_1 = {x1 ↦ x2 ↦ x3 | (x1=FALSE ∧ x2∈BOOL ∧ x3∈BOOL) ∨
(x1∈BOOL ∧ x2=FALSE ∧ x3=FALSE)}
axm7: a_1_1 = InA\(a_1_2)
axm8: a_2_2 = InA\(a_2_3 ∪ a_2_4 ∪ a_234_1)
axm9: a_3_3 = InA\(a_234_1)
axm10: a_4_4 = InA\(a_234_1)
axm11: dta ∈ (A × InA) → A
axm12: ∀s,i•((s=A1 ∧ i∈a_1_1) ∨ (s∈{A2,A3,A4} ∧ i∈a_234_1)) ⇔ dta(s
↦ i) = A1
axm13: ∀s,i•((s=A1 ∧ i∈a_1_2) ∨ (s=A2 ∧ i∈a_2_2)) ⇔ dta(s ↦ i) = A2
axm14: ∀s,i•((s=A2 ∧ i∈a_2_3) ∨ (s=A3 ∧ i∈a_3_3)) ⇔ dta(s ↦ i) = A3
axm15: ∀s,i•((s=A2 ∧ i∈a_2_4) ∨ (s=A4 ∧ i∈a_4_4)) ⇔ dta(s ↦ i) = A4
axm16: LA_y1 ∈ A → BOOL
axm17: LA_y2 ∈ A → BOOL
axm18: LA_y3 ∈ A → BOOL
axm19: LA_y4 ∈ A → BOOL
axm20: ∀s•(s=A2 ∨ s=A3) ⇔ LA_y1(s) = TRUE
axm21: ∀s•(s=A2 ∨ s=A4) ⇔ LA_y2(s) = TRUE
axm22: ∀s•(s=A2) ⇔ LA_y3(s) = TRUE
axm23: ∀s•(s=A3) ⇔ LA_y4(s) = TRUE
```

Динамическая модель алгоритма и окружения

Аксиомы контекста задают статическую часть модели, машина задает динамическую часть [2]. Машина моделирует поведение СВА. В машине объявлены переменные и событие *iter* (см. листинг 3), изменяющее состояние машины. Событие может возникнуть с любыми параметрами x1...x5, удовлетворяющими охранным условиям grd1...grd7. Охранные условия grd1...grd7 моделируют поведение окружения автоматов. Условия grd1...grd5 ограничивают тип параметров x1...x5. СВА имеет пять входов булевого типа. Условия grd6, grd7 ограничивают значения параметров x2, x3. Данные ограничения обусловлены работой блоков f_RzG и f_RzK2. Событие *iter* изменяет состояние машины, выполняя действия act1...act20. Действие act1 изменяет переменную a – текущее состояние автомата A, вызывая его функцию переходов dta. Действия act2...act5 изменяют выходы автомата A, вызывая его функции выходов LA_y1...LA_y4. Действие act6 изменяет переменную b – текущее состояние автомата B, вызывая его функцию переходов dtb. Действия act7...act15 изменяют выходы автомата B, вызывая его функции выходов LB_y1...LB_y9. Действия act16...act20 сохраняют значения параметров x1...x5 в соответствующие им переменные.

Формализация требований

Требования надёжности и безопасности *REQ1*...*REQ16* [1] в виде математических утверждений *T1*...*T16* записываются в разделе инвариантов машины (см. листинг 4) и должны быть доказаны как теоремы.

Теоремы *T1*...*T5* утверждают, что указанные клапаны никогда не открыты одновременно (*REQ1*...*REQ5*).

Теорема *T6* утверждает, что для любого клапана v, принадлежащего множеству клапанов Valve = $\{v_i \mid i=1...10\}$, при снятии сигнала Run клапан v закрывается (*REQ6*).

Теоремы *T7*, *T8* утверждают, что компрессоры работают только при наличии соответствующих разрешений (*REQ7*, *REQ8*).

Теорема *T9* утверждает, что при работе компрессора K2 клапан V2 всегда закрыт – это гарантирует, что воздушный компрессор K2 перекачивает только воздух (*REQ9*).

Теорема *T10* утверждает, что при работе компрессора K1 всегда открыт только один из клапанов V1 или V2 – это гарантирует, что компрессор K1

перекачивает или газ, или воздух, но не их вместе (*REQ10*).

Теорема *T11* утверждает, что если компрессоры К1 и К2 работают, то клапан V2 закрыт – это гарантирует, что компрессоры никогда не перекачивают одно и то же вещество (*REQ11*).

Теорема *T12* утверждает, что снятие сигнала Run отключает компрессоры К1 и К2 (*REQ12*).

Теорема *T13* утверждает, что клапаны V7 и V8 никогда не открыты одновременно – это гарантирует, что эндогаз не может подаваться в холодильник из двух реторт одновременно (*REQ13*).

Теорема *T14_1* утверждает, что для реторты R1 никогда не открыт ни один из воздушных трактов к коллектору. Существует всего три тракта: тракт 1 – при открытии клапанов V7 и V9, тракт 2 – при открытии клапанов V7 и V4, тракт 3 – при открытии клапанов V7, V3, V2. Теорема *T14_2* содержит аналогичное утверждение для реторты R2. Вместе теоремы *T14_1* и *T14_2* гарантируют, что в линии Л3 никогда нет воздуха (*REQ14*).

Теорема *T15* утверждает, что при подаче газа (работает компрессор К1 и открыт клапан V1) обязательно открыт газовый тракт через реторту (*REQ15*). Газовый тракт через реторту R1 возникает при открытии клапана V3 вместе с одним из клапанов V7, V9, через реторту R2 – при открытии клапана V5 вместе с одним из клапанов V8, V10.

Теорема *T16* утверждает, что газ подаётся (открыт клапан V1 и работает компрессор К1) на две реторты (открыты клапаны V3 и V5) и одна из них работает на холодильник (открыт клапан V7 или V8) тогда и только тогда, когда одна реторта находится в рабочем режиме, а вторая в режиме продувки газом (автомат B находится в состоянии B6 или B8). Теорема *T16* гарантирует выполнение требования *REQ16*.

Доказательство теорем

Теоремы *T1...T16* были доказаны в интерактивном режиме. Дерево доказательства теоремы *T16* представлено на рисунке 1. Основные леммы теоремы *T16* представлены в таблице 2. Доказательство имеет следующую стратегию: эквиваленция теоремы разбивается на две импликации *lem1* и *lem2* (T16 = lem1 ∧ lem2). Левая часть импликации леммы *lem1* добавляется к гипотезам и доказывается её правая часть: что функция dtb при принятых гипотезах возвращает значение B6 или B8. Лемма *lem1.1* используется для доказательства леммы *lem1*. Лемма *lem1.1* утверждает, что функция dtb при принятых гипотезах никогда не возвращает значения B1, B2, B3, B4, B5, B7, B9. В итоге из всех возможных значений функции dtb остаётся только два: B6 или B8, что и требовалось доказать в лемме *lem1*.

Левая часть импликации леммы *lem2* добавляется к гипотезам, и доказывается её правая часть: что указанные функции выходов при принятых гипотезах имеют значения TRUE. Доказательство леммы *lem2* разбивается на доказательство ряда лемм: *lem2.1...lem2.5*.

Лемма *lem2.1* утверждает, что функция выходов LA_y3 имеет значение TRUE. Лемма *lem2.1.1* используется для

Листинг 2

```
axm24: partition(B, {B1},{B2},{B3},{B4},{B5},{B6},{B7},{B8},{B9})
axm25: InB = (A × BOOL × BOOL)
axm26: b_1_4 = {x1 ↦ x2 ↦ x3 | x1=A2 ∧ x2=TRUE ∧ x3∈BOOL}
axm27: b_9_8 = b_1_4
axm28: b_1_5 = {x1 ↦ x2 ↦ x3 | x1=A2 ∧ x2=FALSE ∧ x3∈BOOL}
axm29: b_7_6 = b_1_5
axm30: b_4_7 = {x1 ↦ x2 ↦ x3 | x1=A2 ∧ x2∈BOOL ∧ x3=TRUE}
axm31: b_5_9 = b_4_7
axm32: b_6_9 = b_4_7
axm33: b_8_7 = b_4_7
axm34: b_x_1 = {x1 ↦ x2 ↦ x3 | x1=A1 ∧ x2∈BOOL ∧ x3∈BOOL}
axm35: b_x_2 = {x1 ↦ x2 ↦ x3 | x1=A3 ∧ x2∈BOOL ∧ x3∈BOOL}
axm36: b_x_3 = {x1 ↦ x2 ↦ x3 | x1=A4 ∧ x2∈BOOL ∧ x3∈BOOL}
axm37: b_1_1 = InB\(b_1_4 ∪ b_1_5)
axm38: b_2_2 = InB\(b_x_1)
axm39: b_3_3 = InB\(b_x_1)
axm40: b_4_4 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_4_7)
axm41: b_5_5 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_5_9)
axm42: b_6_6 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_6_9)
axm43: b_7_7 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_7_6)
axm44: b_8_8 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_8_7)
axm45: b_9_9 = InB\(b_x_1 ∪ b_x_2 ∪ b_x_3 ∪ b_9_8)
axm46: dtb ∈ (B × InB) → B
axm47: ∀s,i•((s=B1 ∧ i∈b_1_1) ∨ (s∈B\{B1} ∧ i∈b_x_1)) ⇔ dtb(s ↦ i)
= B1
axm48: ∀s,i•((s=B2 ∧ i∈b_2_2) ∨ (s∈B\{B1,B2,B3} ∧ i∈b_x_2)) ⇔ dtb(s
↦ i) = B2
axm49: ∀s,i•((s=B3 ∧ i∈b_3_3) ∨ (s∈B\{B1,B2,B3} ∧ i∈b_x_3)) ⇔ dtb(s
↦ i) = B3
axm50: ∀s,i•((s=B4 ∧ i∈b_4_4) ∨ (s=B1 ∧ i∈b_1_4)) ⇔ dtb(s ↦ i) = B4
axm51: ∀s,i•((s=B5 ∧ i∈b_5_5) ∨ (s=B1 ∧ i∈b_1_5)) ⇔ dtb(s ↦ i) = B5
axm52: ∀s,i•((s=B6 ∧ i∈b_6_6) ∨ (s=B7 ∧ i∈b_7_6)) ⇔ dtb(s ↦ i) = B6
axm53: ∀s,i•((s=B7 ∧ i∈b_7_7) ∨ (s=B4 ∧ i∈b_4_7) ∨ (s=B8 ∧ i∈b_8_7))
⇔ dtb(s ↦ i) = B7
axm54: ∀s,i•((s=B8 ∧ i∈b_8_8) ∨ (s=B9 ∧ i∈b_9_8)) ⇔ dtb(s ↦ i) = B8
axm55: ∀s,i•((s=B9 ∧ i∈b_9_9) ∨ (s=B5 ∧ i∈b_5_9) ∨ (s=B6 ∧ i∈b_6_9))
⇔ dtb(s ↦ i) = B9
axm56: LB_y1 ∈ B → BOOL
axm57: LB_y2 ∈ B → BOOL
axm58: LB_y3 ∈ B → BOOL
axm59: LB_y4 ∈ B → BOOL
axm60: LB_y5 ∈ B →BOOL
axm61: LB_y6 ∈ B → BOOL
axm62: LB_y7 ∈ B → BOOL
axm63: LB_y8 ∈ B → BOOL
axm64: LB_y9 ∈ B → BOOL
axm65: ∀s•s∈{B2,B4,B6,B7,B8} ⇔ LB_y1(s) = TRUE
axm66: ∀s•s∈{B3,B5,B9} ⇔ LB_y2(s) = TRUE
axm67: ∀s•s∈{B2,B5,B6,B8,B9} ⇔ LB_y3(s) = TRUE
axm68: ∀s•s∈{B3,B4,B7} ⇔ LB_y4(s) = TRUE
axm69: ∀s•s∈{B6,B7} ⇔ LB_y5(s) = TRUE
axm70: ∀s•s∈{B8,B9} ⇔ LB_y6(s) = TRUE
axm71: ∀s•s∈{B2,B3,B4,B5,B8,B9} ⇔ LB_y7(s) = TRUE
axm72: ∀s•s∈{B2,B3,B4,B5,B6,B7} ⇔ LB_y8(s) = TRUE
axm73: ∀s•s∈{B4,B5,B6,B8} ⇔ LB_y9(s) = TRUE
```

Таблица 1. Назначение аксиом

Аксиома	Назначение (что определяет аксиома)
axm1	A – множество состояний автомата A
axm2	InA – множество всех векторов входных сигналов автомата A
axm3… axm10	a_i_j – множество векторов входных сигналов автомата A для перехода из состояния i в состояние j
axm11	Тип функции переходов автомата A
axm12… axm15	dta – функция переходов автомата A
axm16… axm19	Тип функции выходов автомата A
axm20… axm23	LA – функция выходов автомата A
axm24	B – множество состояний автомата B
axm25	InB – множество всех векторов входных сигналов автомата B
axm26… axm45	b_i_j – множество векторов входных сигналов автомата B для перехода из состояния i в состояние j
axm46	Тип функции переходов автомата B
axm47… axm55	dtb – функция переходов автомата B
axm56… axm64	Тип функции выходов автомата B
axm65… axm73	LB – функция выходов автомата B

Листинг 3

```
iter
ANY
 x1
 x2
 x3
 x4
 x5
WHERE
 grd1: x1 ∈ BOOL
 grd2: x2 ∈ BOOL
 grd3: x3 ∈ BOOL
 grd4: x4 ∈ BOOL
 grd5: x5 ∈ BOOL
 grd6: (a = A3) ⇒ (x2 = TRUE ∧ x3 = FALSE)
 grd7: (a = A4) ⇒ (x2 = FALSE ∧ x3 = TRUE)
THEN
 act1: a ≔ dta(a ↦ (x1 ↦ x2 ↦ x3))
 act2: K1 ≔ LA_y1(dta(a ↦ (x1 ↦ x2 ↦ x3)))
 act3: K2 ≔ LA_y2(dta(a ↦ (x1 ↦ x2 ↦ x3)))
 act4: V1 ≔ LA_y3(dta(a ↦ (x1 ↦ x2 ↦ x3)))
 act5: V2 ≔ LA_y4(dta(a ↦ (x1 ↦ x2 ↦ x3)))
 act6: b ≔ dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))
 act7: V3 ≔ LB_y1(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act8: V4 ≔ LB_y2(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act9: V5 ≔ LB_y3(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act10: V6 ≔ LB_y4(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act11: V7 ≔ LB_y5(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act12: V8 ≔ LB_y6(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act13: V9 ≔ LB_y7(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act14: V10 ≔ LB_y8(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act15: tmrQ ≔ LB_y9(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))
 act16: Run ≔ x1
 act17: RzG ≔ x2
 act18: RzK2 ≔ x3
 act19: sw1 ≔ x4
 act20: tmrI ≔ x5
END
```

Листинг 4

```
INVARIANTS
...
T1: ¬(V1=TRUE ∧ V2=TRUE)
T2: ¬(V3=TRUE ∧ V4=TRUE)
T3: ¬(V5=TRUE ∧ V6=TRUE)
T4: ¬(V7=TRUE ∧ V9=TRUE)
T5: ¬(V8=TRUE ∧ V10=TRUE)
T6: ∀v•(v∈Valve ∧ Run=FALSE) ⇒ closed(v)=TRUE
T7: K1=TRUE ⇒ RzG=TRUE
T8: K2=TRUE ⇒ RzK2=TRUE
T9: K2=TRUE ⇒ V2=FALSE
T10: K1=TRUE ⇒ ((V1=TRUE ∧ V2=FALSE) ∨ (V2=TRUE ∧ V1=FALSE))
T11: (K1=TRUE ∧ K2=TRUE) ⇒ V2=FALSE
T12: (Run=FALSE) ⇒ (K1=FALSE ∧ K2=FALSE)
T13: ¬(V7=TRUE ∧ V8=TRUE)
T14_1: ¬((V7=TRUE ∧ V9=TRUE) ∨ (V7=TRUE ∧ V4=TRUE) ∨ (V7=TRUE ∧
V3=TRUE ∧ V2=TRUE))
T14_2: ¬((V8=TRUE ∧ V10=TRUE) ∨ (V8=TRUE ∧ V6=TRUE) ∨ (V8=TRUE ∧
V5=TRUE ∧ V2=TRUE))
T15: (K1=TRUE ∧ V1=TRUE) ⇒ ((V3=TRUE ∧ (V7=TRUE ∨ V9=TRUE)) ∨
(V5=TRUE ∧ (V8=TRUE ∨ V10=TRUE)))
T16: (V1=TRUE ∧ K1=TRUE ∧ V3=TRUE ∧ V5=TRUE ∧ (V7=TRUE ∨ V8=TRUE)) ⇔
(b=B6 ∨ b=B8)
```

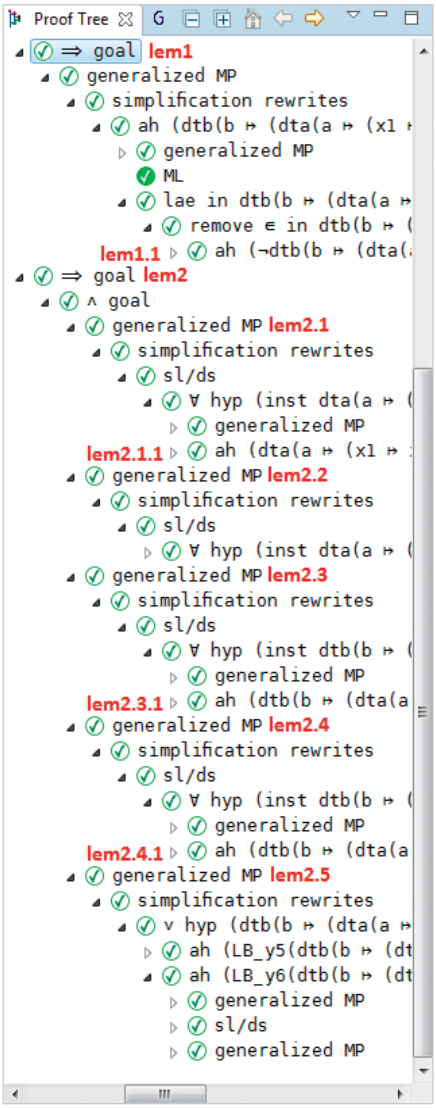

Рис. 1. Дерево доказательства теоремы T16

доказательства леммы *lem2.1*. Лемма *lem2.1.1* утверждает, что функция dta при принятых гипотезах всегда возвращает значение A2. Доказательство леммы *lem2.1.1* основывается на том, что если функция dtb возвращает значение B6 или B8, то функция dta не может вернуть ничего кроме A2. Если автомат A находится в состоянии A2, то LA_y3 имеет значение TRUE (согласно аксиоме *axm22*), что и требовалось доказать в лемме *lem2.1*.

Лемма *lem2.2* утверждает, что функция выходов LA_y1 имеет значение TRUE. Для доказательства этой леммы используется уже доказанная лемма *lem2.1.1* и аксиома *axm20*.

Лемма *lem2.3* утверждает, что функция выходов LB_y1 имеет значение TRUE. Лемма *lem2.3.1* используется для доказательства леммы *lem2.3*. Лемма *lem2.3.1* утверждает, что функция dtb при принятых гипотезах всегда возвращает значения, принадлежащие множеству {B2, B4, B6, B7, B8}. Если автомат B находится в одном из перечисленных состояний, то LB_y1 имеет значение TRUE (согласно аксиоме *axm65*), что и требовалось доказать в лемме *lem2.3*. Лемма *lem2.4* имеет аналогичное доказательство.

Лемма *lem2.5* утверждает, что функция выходов LB_y5 или LB_y6 имеет значение TRUE. В ветке доказательства леммы *lem2* принята следующая гипотеза: функция dtb возвращает значение B6 или B8. Если функция dtb возвращает значение B6, то функция LB_y5 имеет значение TRUE (согласно аксиоме *axm69*), если функция dtb возвращает значение B8, то функция LB_y6 имеет значение TRUE (согласно аксиоме *axm70*), что и требовалось доказать в лемме *lem2.5*. Таким образом, доказательство всех перечисленных лемм доказывает теорему *T16*. Леммы *lem1.1*, *lem2.1.1*, *lem2.3.1*, *lem2.4.1* были добавлены в дерево доказательства вручную, остальные леммы таблицы 2 были сгенерированы программой-прувером автоматически.

Заключение

В статье рассмотрен алгоритм управления генератором эндогаза, спроектированный как СВА. Для обеспечения надёжности и безопасности технологического процесса алгоритм должен гарантированно соответствовать ряду требований. Для доказательства соответствия требованиям выполняется формальный дедуктивный анализ. С помощью платформы Rodin строится математическая модель алгоритма и его окружения. Требования подлежат формализации. Каждое требование задаётся как инвариант машины,

Таблица 2. Основные леммы теоремы T16

Лемма	Утверждение
lem1	LA_y3(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE ∧ LA_y1(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE ∧ LB_y1(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∧ LB_y3(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∧ (LB_y5(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∨ LB_y6(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE) ⇒ (dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))=B6 ∨ dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))=B8) Интерпретация: V1=TRUE ∧ K1=TRUE ∧ V3=TRUE ∧ V5=TRUE ∧ (V7=TRUE ∨ V8=TRUE) ⇒ (b=B6 ∨ b=B8)
lem1.1	¬dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))∈{B1,B2,B3,B4,B5, B7,B9}
lem2	(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))=B6 ∨ dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))=B8) ⇒ LA_y3(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE ∧ LA_y1(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE ∧ LB_y1(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∧ LB_y3(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∧ (LB_y5(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∨ LB_y6(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE) Интерпретация: (b=B6 ∨ b=B8) ⇒ V1=TRUE ∧ K1=TRUE ∧ V3=TRUE ∧ V5=TRUE ∧ (V7=TRUE ∨ V8=TRUE)
lem2.1	LA_y3(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE Интерпретация: V1=TRUE
lem2.1.1	dta(a ↦ (x1 ↦ x2 ↦ x3))=A2
lem2.2	LA_y1(dta(a ↦ (x1 ↦ x2 ↦ x3)))=TRUE Интерпретация: K1=TRUE
lem2.3	LB_y1(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE Интерпретация: V3=TRUE
lem2.3.1	dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))∈{B2,B4,B6,B7,B8}
lem2.4	LB_y3(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE Интерпретация: V5=TRUE
lem2.4.1	dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5))∈{B2,B5,B6,B8,B9}
lem2.5	(LB_y5(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE ∨ LB_y6(dtb(b ↦ (dta(a ↦ (x1 ↦ x2 ↦ x3)) ↦ x4 ↦ x5)))=TRUE) Интерпретация: (V7=TRUE ∨ V8=TRUE)

моделирующей поведение. Инварианты доказываются как теоремы. Доказательство теорем гарантирует соответствие требованиям. Платформа Rodin ускоряет и упрощает доказательства теорем, а также исключает их ошибочность.

Формальный дедуктивный анализ хорошо сочетается с автоматным проектированием алгоритмов: для формализации алгоритма не требуется дополнительных преобразований, так как автоматы уже представляют собой математические объекты.

Литература

1. *Нейзов М.* Формальный дедуктивный анализ автоматного алгоритма управления генератором эндогаза с помощью платформы Rodin. Часть 1. Определение требований надежности и безопасности работы генератора эндогаза. Современная электроника. 2020. № 9.
2. *Нейзов М.* Формальный дедуктивный анализ автоматного алгоритма управления генератором эндогаза с помощью платформы Rodin. Часть 2. Алгоритм управления и платформа Rodin. Современная электроника. 2021. № 1.