Ingénierie dirigée par les modèles assistée par les Large Language Models (LLM)
Automated System Engineering with Artificial Intelligence

SystemX - 2024

Ludovic Apvrille
ludovic.apvrille@telecom-paris.fr

Outline

Introduction

Context

Contribution overview

Application to System Engineering

Evaluation

Demo

Conclusion

Introduction

LLM and System Engineering

- AI is now used for many applications
- **Could LLM (e.g., ChatGPT) be used to assist system architects?**
    - Modeling, verification, coding

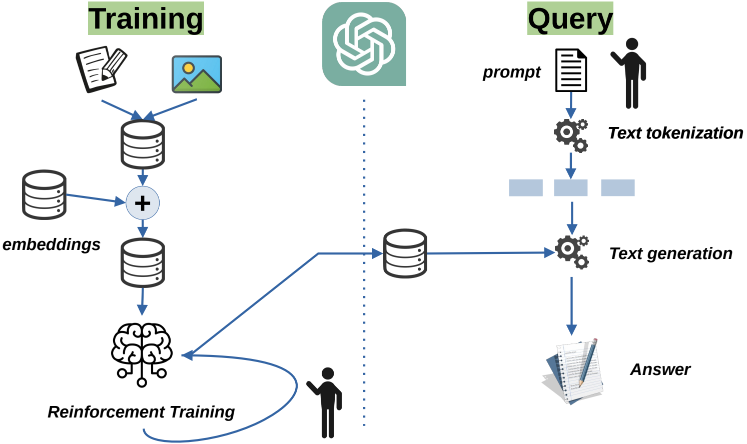
Outline

- Context: methodologies for designing embedded systems, ChatGPT (and similar AI)
- Contribution on S.E.: general approach, implementation aspects, evaluation
- Demo, conclusion

Outline

- Introduction
- Context
- Contribution overview
- Application to System Engineering
- Evaluation
- Demo
- Conclusion

ChatGPT in a Nutshell



https://writings.stephenwolfram.com/2023/02/what-is-chatgpt-doing-and-why-does-it-work/

Our Idea: AI&ChatGPT to Support System Engineering

(Model-Based) System Engineering = document understanding and transformation into views/diagrams/code

Input

- Improve ChatGPT to understand inputs such as:
    - System specification
    - Objective of the request: identifying requirement, identifying system blocks, etc.
    - Constraints from system architects
    - Existing diagrams or code

Output

- Drive ChatGPT to produce answers related to system engineering
- Drive ChatGPT to produce an easy-to-parse output
- Automatically identify if produced outputs are invalid/wrong, and force ChatGPT to produce better answers

Outline

Introduction

Context

Contribution overview

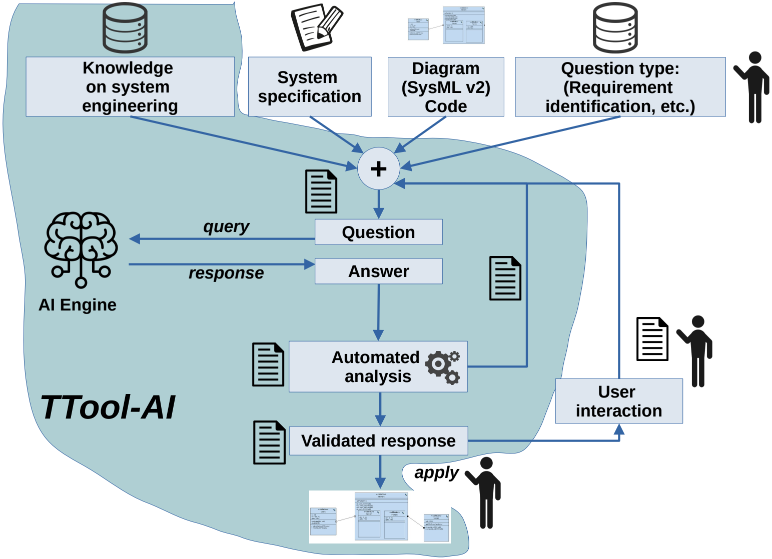
Application to System Engineering

Evaluation

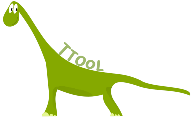
Demo

Conclusion

Contribution: Overview



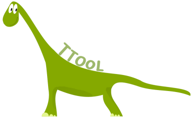
TTool

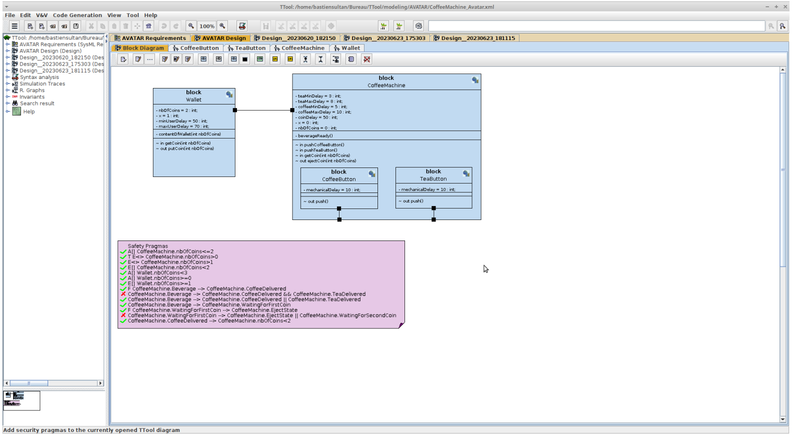


An Open-Source Toolkit

- Formal system modeling in SysML
  - Views: fonctional, architecture, deployment, system, etc.
  - Diagrams: requirement, use case, block, state machines, . . .
- Support for all system development cycle (from requirements to code)
- Safety / security /performance evaluation at the push of a button
- Code generation
- Industrial and academic contribution and use
- `ttool.telecom-paris.fr`

TTool



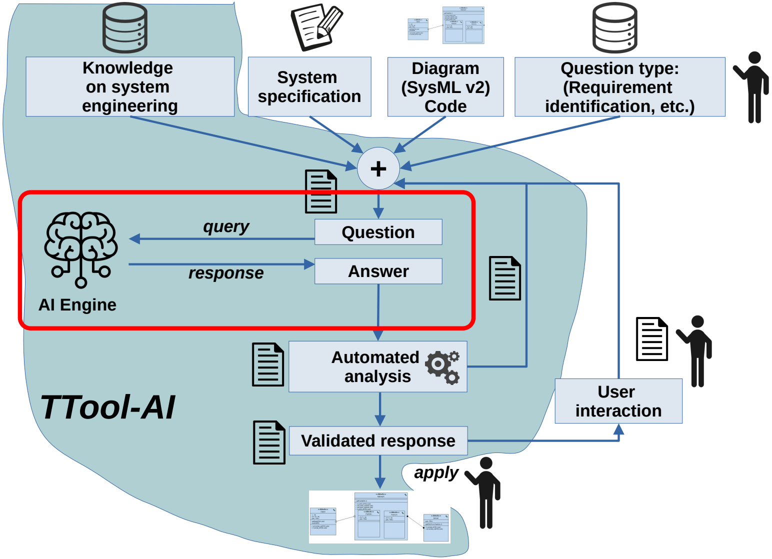


Safety Pragmas

```
✓ A[] CoffeeMachine.nbOfCoins<=2
✓ T E<> CoffeeMachine.nbOfCoins>0
✓ E<> CoffeeMachine.nbOfCoins>1
✓ E[] CoffeeMachine.nbOfCoins<2
✓ A[] Wallet.nbOfCoins<3
✓ A[] Wallet.nbOfCoins>=0
✓ E[] Wallet.nbOfCoins>=1
✓ F CoffeeMachine.Beverage --> CoffeeMachine.CoffeeDelivered
✗ CoffeeMachine.Beverage --> CoffeeMachine.CoffeeDelivered && CoffeeMachine.TeaDelivered
✓ CoffeeMachine.Beverage --> CoffeeMachine.CoffeeDelivered || CoffeeMachine.TeaDelivered
✓ CoffeeMachine.Beverage --> CoffeeMachine.WaitingForFirstCoin
✓ F CoffeeMachine.WaitingForFirstCoin --> CoffeeMachine.EjectState
✗ CoffeeMachine.WaitingForFirstCoin --> CoffeeMachine.EjectState || CoffeeMachine.WaitingForSecondCoin
✓ CoffeeMachine.CoffeeDelivered --> CoffeeMachine.nbOfCoins<2
```

Add security pragmas to the currently opened TTool diagram

Contribution: Interacting with ChatGPT



Knowledge on system engineering

System specification

Diagram (SysML v2) Code

Question type: (Requirement identification, etc.)

+

query

response

Question

Answer

AI Engine

Automated analysis

TTool-AI

User interaction

Validated response

apply

ChatGPT API: the Basic Way

```
$ curl https://api.openai.com/v1/chat/completions
-H "Content-Type: application/json"
-H "Authorization: Bearer OPENAI_API_KEY"
-d '{ "model": "gpt-4o",
  "messages": [{"role": "user", "content": "Hello!"}]
}'

{ "id": "chatcmpl-7YzJ6Gmh0VjbWl3plUxqwzDrtDnnk",
"object": "chat.completion", "created": 1688573172,
"model": "gpt-4o", "choices": [ { "index": 0,
"message": { "role": "assistant",
"content": "Hello! How can I assist you today?" },
"finish_reason": "stop" } ], \
"usage": { "prompt_tokens": 8, "completion_tokens": 9,
"total_tokens": 17 } }
```

You first need to buy tokens to get a valid OPENAI_API_KEY

ChatGPT API: Still quite Basic

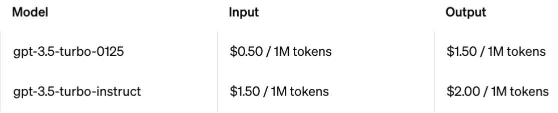
```
$ curl https://api.openai.com/v1/chat/completions
  -H "Content-Type: application/json"
  -H "Authorization: Bearer OPENAI_API_KEY"
  -d '{ "model": "gpt-3.5-turbo",
    "messages": [{"role": "user", "content": "Hello!"}]
  }'| jq '.choices[]'.message.content

Hello! How can I assist you today?

$ chatgpt "Is IRT SystemX a good research institute?"

"The IRT SystemX, also known as Institut de Recherche Technologique
SystemX, is a renowned research and technology organization located
in France. It focuses on the development of innovative solutions in
various fields such as digital transformation, complex systems
engineering, and intelligent systems. Here are some factors to consider
when evaluating its reputation as a research institute ..."
```

ChatGPT API: Pricing

Token-based

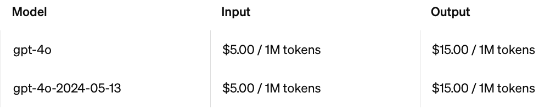
- Input / output tokens
- Limits on the number of queries per unit of time

ChatGPT 3.5



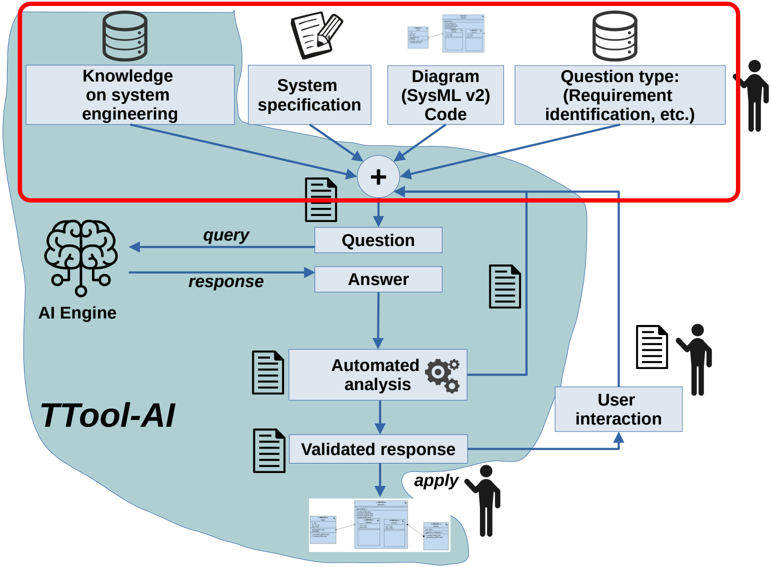
Model	Input	Output
gpt-3.5-turbo-0125	$0.50 / 1M tokens	$1.50 / 1M tokens
gpt-3.5-turbo-instruct	$1.50 / 1M tokens	$2.00 / 1M tokens

ChatGPT 4



Model	Input	Output
gpt-4o	$5.00 / 1M tokens	$15.00 / 1M tokens
gpt-4o-2024-05-13	$5.00 / 1M tokens	$15.00 / 1M tokens

Contribution: Custom Knowledge (or Context)



Knowledge on system engineering

System specification

Diagram (SysML v2) Code

Question type: (Requirement identification, etc.)

+

Question

query

response

Answer

AI Engine

Automated analysis

User interaction

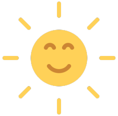
Validated response

apply

TTool-AI

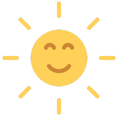
ChatGPT API: Custom Knowledge (or Context) (Cont.)

```
$ curl https://api.openai.com/v1/chat/completions
...
"messages": [{"role": "user", "content": "Is Sophia-Antipolis sunny today?"}]
...

"I m sorry, as an AI language model, I dont have access to current weather
conditions. However, Sophia-Antipolis is located in the French Riviera,
which has a generally sunny and Mediterranean climate..."
```

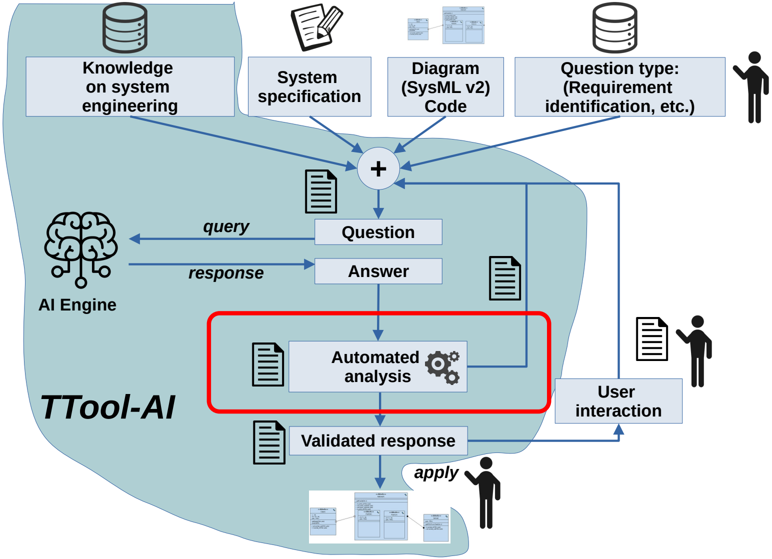


ChatGPT API: Custom Knowledge (or Context) (Cont.)

```
$ curl https://api.openai.com/v1/chat/completions \
...
{"messages":[{"role":"user","content":"Today, in Sophia-Antipolis,
the sun shines with no clouds."},{"role":"assistant","content":"ok"},
{"role":"user","content":"Is Sophia-Antipolis sunny today?"}]
...

"Yes, you mentioned earlier that the sun is shining with no clouds
in Sophia-Antipolis. So, it is indeed sunny today in
Sophia-Antipolis. Enjoy the beautiful weather!"
```



Contribution: Automated Answer Analysis



Knowledge on system engineering

System specification

Diagram (SysML v2) Code

Question type: (Requirement identification, etc.)

+

query

Question

response

Answer

AI Engine

Automated analysis

User interaction

TTool-AI

Validated response

apply

Automated Answer Analysis: Objectives

Invalid answer

Answer that does not respect the expected output:

- Invalid format (JSON, XML, SysML v2, . . .)
- Constraints not respected
  - "List two blocks in json format", but the answer gives only one block

Detection and iteration

1. Detect format or constraint errors in answer
2. If at least one error is detected and $cpt < maxIteration$ not reached
   - 2.1 The previous message and its answer are added to the context
   - 2.2 A description d_i is generated for each error e_i
   - 2.3 A new message is formed based on the list of d_i
   - 2.4 The message is fed into ChatGPT
   - 2.5 $cpt++$; Goto 1

Automated Answer Analysis: Example

```
$ curl https://api.openai.com/v1/chat/completions
...
"messages":[
{"role":"system",
"content":"You are a helpful assistant for
system engineering."},

{"role":"user",
"content":"Give me the example of a SysML block, in json
format, with attributes. No attribute of type String. The block should
have one more attribute than ports, and one less port than operations.
The number of ports must be equal to 1."}]
...
```

Automated Answer Analysis: Example (Cont.)

```
{ "name": "Block1", "attributes": [
    {"name": "attribute1", "type": "Integer"},
    {"name": "attribute2", "type": "Boolean"},
    {"name": "attribute3", "type": "Float"},
    {"name": "attribute4", "type": "Double"}],
  "ports": [
    {"name": "port1","type": "Port1"}],
  "operations": [
    {"name": "operation1", "parameters": [], "returnType": "ReturnType1"},
    {"name": "operation2", "parameters": [], "returnType": "ReturnType2"}
  ]
}
```

Automated Answer Analysis: Example (Cont.)

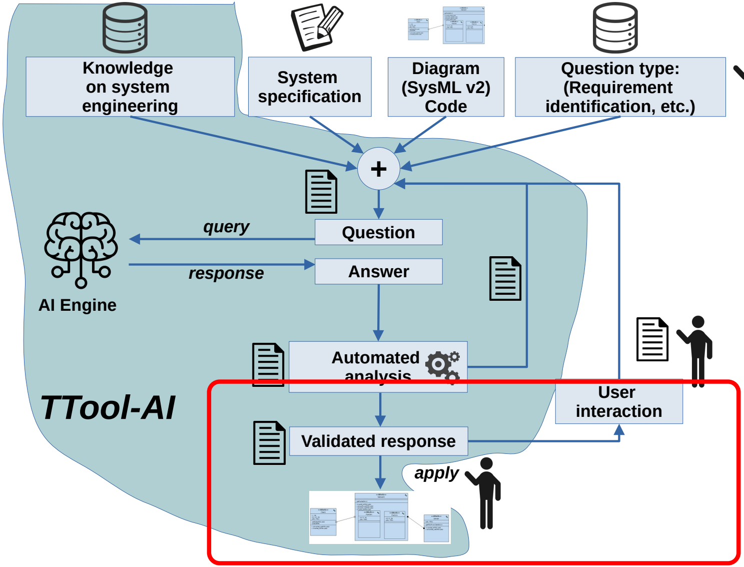
We compute an updated question and we send it:

```
$ curl https://api.openai.com/v1/chat/completions \
...
"messages":[
{"role":"system",
"content":"You are a helpful assistant for
system engineering."},

{"role":"user",
"content":"Give me the example of a SysML block, in json
format, with attributes. No attribute of type String....
The number of ports must be equal to 1."},

{"role":"assistant",
"content": { "name": "Block1", "attributes": [...]},

{"role":"user",
"content":"Your answer does not respect the following constraint:
The block should have one more attribute than ports.
Update your answer accordingly."}]
...
```

Automated Answer Analysis: Example (Cont.)

```
{ "name": "Block1", "attributes": [
    {"name": "attribute1", "type": "Integer"},
    {"name": "attribute2", "type": "Boolean"},
  "ports": [
    {"name": "port1","type": "Port1"}],
  "operations": [
    {"name": "operation1", "parameters": [], "returnType": "ReturnType1"},
    {"name": "operation2", "parameters": [], "returnType": "ReturnType2"}
  ]
}
```

Contribution: User Interaction and Result Displaying



Outline

Introduction

Context

Contribution overview

Application to System Engineering

Evaluation

Demo

Conclusion

General Contribution Applied to System Engineering

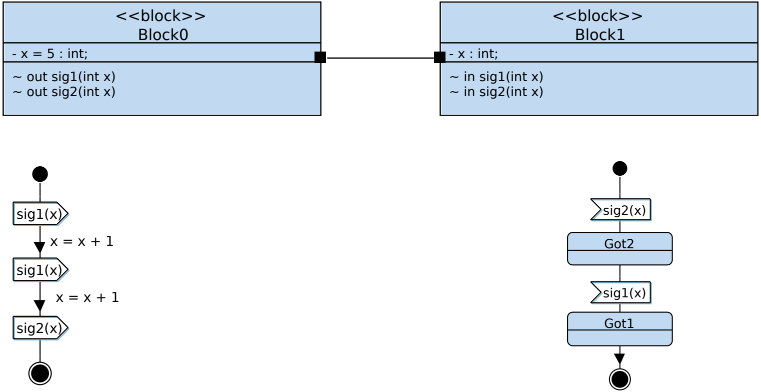
Objective: efficiently use the presented framework for automated system engineering

- Define a context (knowledge in system engineering, output format, expected output (a diagram, a list of elements, etc.))
- Define a list of questions linked with System Engineering
    - Requirement identification, requirement classification, . . .
- Mandatory / optional inputs for questions:
    - System specification (text) - user provided
    - Diagram (in SysML V2 format)
- Automated analysis of answers, and iterations
- Diagram drawing from output

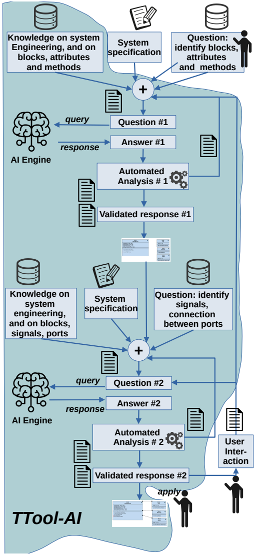
Focus on SysML Design

SysML design

= a set of interconnected blocks (structure) and one state machine for each block (behavior)



Focus on SysML Design: Knowledge Splitting



Context restriction

- Context = (Knowledge + question + specification) must fit in x tokens
- Providing knowledge on blocks, attributes, signals, connections, etc. takes many tokens
    - ⇒ No space left for system specification and question
- Large contexts may also lead to worse results
- Price

Focus on SysML Design: Blocks

Context for the identification of blocks and attributes

```
When you are asked to identify SysML blocks, return them as a
JSON specification formatted as follows:
{blocks: [{ \"name\": \"Name of block\", \"attributes\": [\"name\":
\"name of attribute\", \"type\": \"int or bool\" ...} ...]}

#Respect: each attribute must be of type \"int\" or \"bool\" only
#Respect: Any identifier (block, attribute, etc.) must no contain
any space. Use \"_\" instead.
...
```

Focus on SysML Design: Blocks (Cont.)

Question for the identification of blocks and attributes

```
From the following system specification, using the specified JSON format,
identify the typical system blocks and their attributes. Do respect the
 JSON format, and provide only JSON (no explanation before or after).
```

- Question is concatenated with the user-provided specification and user-provided (extra) constraints
- Automated verification: JSON format, constraints

Similar approach for:

- Identification of connections between blocks
- Identification of state machines

Focus on SysML Design: Blocks (Cont.)

Context for connecting blocks

```
...
#Respect: 2 signals with the same name are assumed to be connected: this
is the only way to connect signals.
#Respect: Two connected signals must have the same list of attributes,
even if they are defined in two different blocks. One of them must be
output, the other one must be input.
#Respect: all input signals must have exactly one corresponding
output signal, i.e., an ouput signal with the same name
...
```

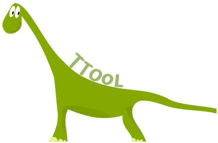
Question for connecting blocks

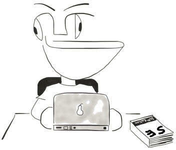
```
From the previous JSON and system specification, update this JSON with
the signals you have to identify. If necessary, you can add new blocks
and new attributes.
```

Outline

- Introduction
- Context
- Contribution overview
- Application to System Engineering
- Evaluation
- Demo
- Conclusion

TTool-AI vs Master-Level Students

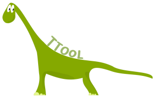


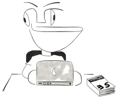


Test setup

- Same system specification is given to TTool-AI and to 15 master-level students after 21h of training
- TTool-AI's models and Student's models are evaluated with the same grading policies
- Comparison on average and standard deviation for grades, and for the time taken to produce models
- Use of ChatGPT 3.5 turbo

TTool-AI vs Master-Level Students (Cont.)





	Time BD (s)	Grade BD (/100)	Time SMD (s)	Grade SMD (/100)
Average	40	81	178	63
Std dev.	10	16	97	15

(a) *TTool-AI*

	Time BD (s)	Grade BD (/100)	Time SMD (s)	Grade SMD (/100)
Average	2700	70	2700	58
Std dev.	—	26	—	32

(b) Students

- Better average for TTool-AI on grades, and it is much faster
- Best students always do better than TTool-AI, but with a much higher time
  - And it is hard to predict which students will be the best for a given specification

Outline

Introduction

Context

Contribution overview

Application to System Engineering

Evaluation

Demo

Conclusion

Demo: a Coffee Machine

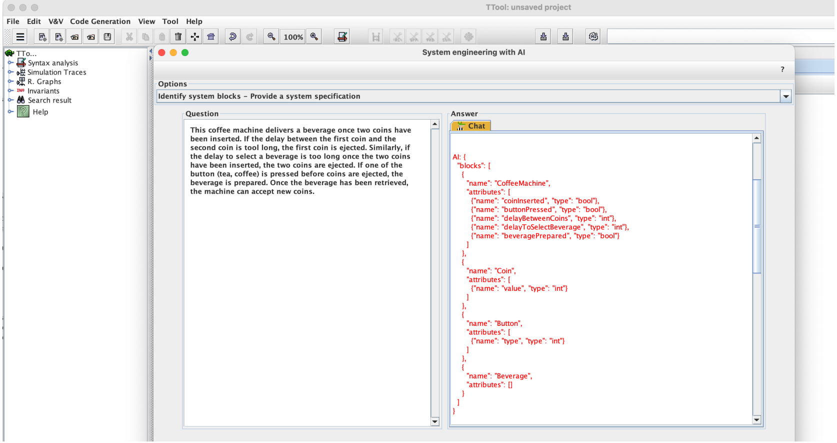
Objective: making the block and state machine diagrams of the following system:

This coffee machine delivers a beverage once two coins have been inserted. If the delay between the first coin and the second coin is too long, the first coin is ejected. Similarly, if the delay to select a beverage is too long once the two coins have been inserted, the two coins are ejected. If one of the button (tea, coffee) is pressed before coins are ejected, the beverage is prepared. Once the beverage has been retrieved, the machine can accept new coins.

TTool is used for query/answer to/from ChatGPT and diagram drawing

`ttool.telecom-paris.fr`

Demo: a Coffee Machine (Cont.)



Identify system blocks – Provide a system specification

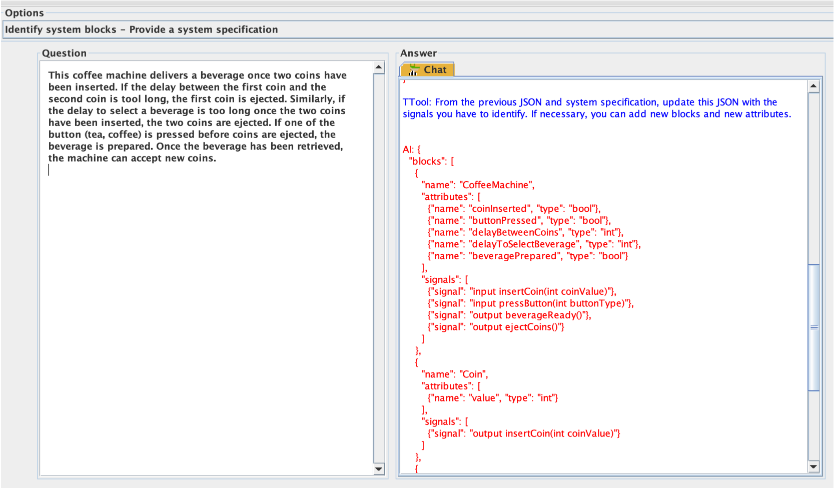
Question

This coffee machine delivers a beverage once two coins have been inserted. If the delay between the first coin and the second coin is tool long, the first coin is ejected. Similarly, if the delay to select a beverage is too long once the two coins have been inserted, the two coins are ejected. If one of the button (tea, coffee) is pressed before coins are ejected, the beverage is prepared. Once the beverage has been retrieved, the machine can accept new coins.

Answer

```
AI: {
  "blocks": [
    {
      "name": "CoffeeMachine",
      "attributes": [
        {"name": "coinInserted", "type": "bool"},
        {"name": "buttonPressed", "type": "bool"},
        {"name": "delayBetweenCoins", "type": "int"},
        {"name": "delayToSelectBeverage", "type": "int"},
        {"name": "beveragePrepared", "type": "bool"}
      ]
    },
    {
      "name": "Coin",
      "attributes": [
        {"name": "value", "type": "int"}
      ]
    },
    {
      "name": "Button",
      "attributes": [
        {"name": "type", "type": "int"}
      ]
    },
    {
      "name": "Beverage",
      "attributes": []
    }
  ]
}
```

Demo: a Coffee Machine (Cont.)

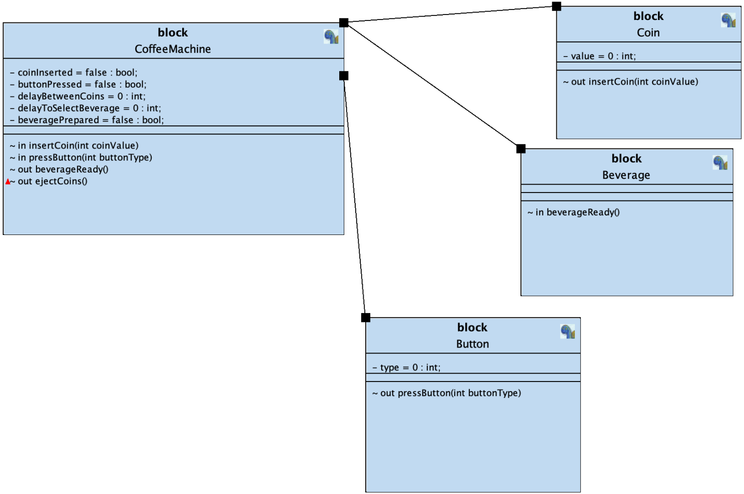


This coffee machine delivers a beverage once two coins have been inserted. If the delay between the first coin and the second coin is tool long, the first coin is ejected. Similarly, if the delay to select a beverage is too long once the two coins have been inserted, the two coins are ejected. If one of the button (tea, coffee) is pressed before coins are ejected, the beverage is prepared. Once the beverage has been retrieved, the machine can accept new coins.

TTool: From the previous JSON and system specification, update this JSON with the signals you have to identify. If necessary, you can add new blocks and new attributes.

```
AI: {
 "blocks": [
  {
    "name": "CoffeeMachine",
    "attributes": [
      {"name": "coinInserted", "type": "bool"},
      {"name": "buttonPressed", "type": "bool"},
      {"name": "delayBetweenCoins", "type": "int"},
      {"name": "delayToSelectBeverage", "type": "int"},
      {"name": "beveragePrepared", "type": "bool"}
    ],
    "signals": [
      {"signal": "input insertCoin(int coinValue)"},
      {"signal": "input pressButton(int buttonType)"},
      {"signal": "output beverageReady()"},
      {"signal": "output ejectCoins()"}
    ]
  },
  {
    "name": "Coin",
    "attributes": [
      {"name": "value", "type": "int"}
    ],
    "signals": [
      {"signal": "output insertCoin(int coinValue)"}
    ]
  },
  {
```

Demo: a Coffee Machine (Cont.)



block
CoffeeMachine

- coinInserted = false : bool;
- buttonPressed = false : bool;
- delayBetweenCoins = 0 : int;
- delayToSelectBeverage = 0 : int;
- beveragePrepared = false : bool;

~ in insertCoin(int coinValue)
~ in pressButton(int buttonType)
~ out beverageReady()
~ out ejectCoins()

block
Coin

- value = 0 : int;

~ out insertCoin(int coinValue)

block
Beverage

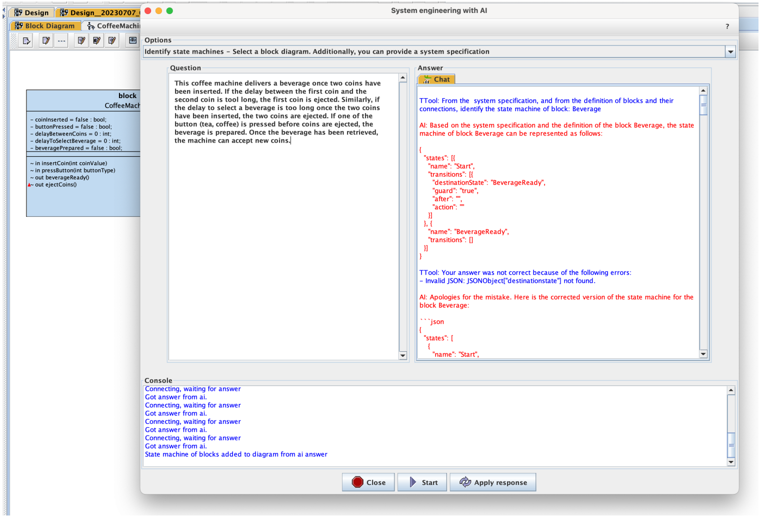
~ in beverageReady()

block
Button

- type = 0 : int;

~ out pressButton(int buttonType)

Demo: a Coffee Machine (Cont.)



```
- coinInserted = false : bool;
- buttonPressed = false : bool;
- delayBetweenCoins = 0 : int;
- delayToSelectBeverage = 0 : int;
- beveragePrepared = false : bool;

~ in insertCoin(int coinValue)
~ in pressButton(int buttonType)
~ out beverageReady()
~ out ejectCoins()
```

Identify state machines - Select a block diagram. Additionally, you can provide a system specification

This coffee machine delivers a beverage once two coins have been inserted. If the delay between the first coin and the second coin is tool long, the first coin is ejected. Similarly, if the delay to select a beverage is too long once the two coins have been inserted, the two coins are ejected. If one of the button (tea, coffee) is pressed before coins are ejected, the beverage is prepared. Once the beverage has been retrieved, the machine can accept new coins.

TTool: From the system specification, and from the definition of blocks and their connections, identify the state machine of block: Beverage

AI: Based on the system specification and the definition of the block Beverage, the state machine of block Beverage can be represented as follows:

```
{
  "states": [{
    "name": "Start",
    "transitions": [{
      "destinationState": "BeverageReady",
      "guard": "true",
      "after": "",
      "action": ""
    }]
  }, {
    "name": "BeverageReady",
    "transitions": []
  }]
}
```

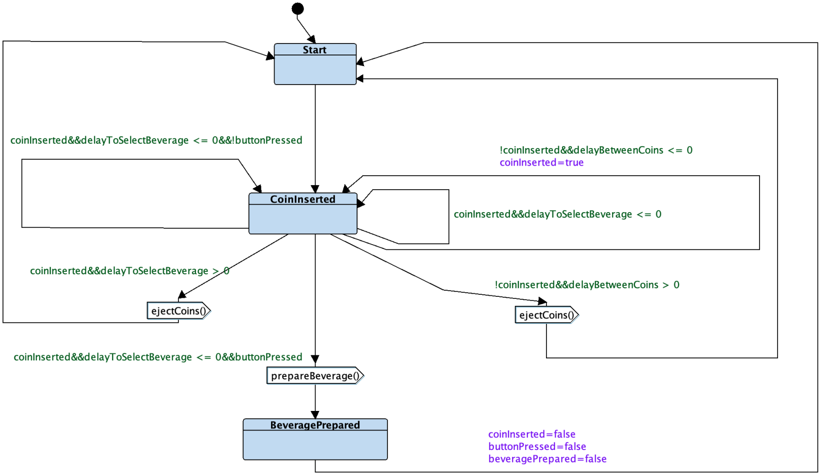
TTool: Your answer was not correct because of the following errors:
- Invalid JSON: JSONObject["destinationstate"] not found.

AI: Apologies for the mistake. Here is the corrected version of the state machine for the block Beverage:

~~~
```json
{
  "states": [
    {
      "name": "Start",
~~~

```
Connecting, waiting for answer
Got answer from ai.
Connecting, waiting for answer
Got answer from ai.
Connecting, waiting for answer
Got answer from ai.
Connecting, waiting for answer
Got answer from ai.
State machine of blocks added to diagram from ai answer
```

Demo: a Coffee Machine (Cont.)


Start
coinInserted&&delayToSelectBeverage <= 0&&!buttonPressed
!coinInserted&&delayBetweenCoins <= 0
coinInserted=true
CoinInserted
coinInserted&&delayToSelectBeverage <= 0
coinInserted&&delayToSelectBeverage > 0
!coinInserted&&delayBetweenCoins > 0
ejectCoins()
ejectCoins()
coinInserted&&delayToSelectBeverage <= 0&&buttonPressed
prepareBeverage()
BeveragePrepared
coinInserted=false
buttonPressed=false
beveragePrepared=false

Outline

Introduction

Context

Contribution overview

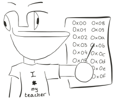
Application to System Engineering

Evaluation

Demo

Conclusion

To Go Further with TTool-AI. . .



ChatGPT is a chatbot: it tells stories related to your questions, elaborating a probable answer, with no guarantees that the answer is correct.

1. **Analysis of system specification is rather impressive**: extracting requirements, use cases, design (structure, behavior) and properties identification
2. **Need for automated and manual supervision**
   - **Trade-off between quality, price and time**
3. **Best students can do better but with a much higher time**
   - **Trade-off between quality and time, and how to pre-identify best students?**
4. **Verifying that a property is true for a given system usually fails** (ChatGPT can be wrong at solving trivial mathematical exercises. . .)
   - ChatGPT refuses to prove properties for complex diagram

Version 4 of ChatGPT is even better, not covered in the evaluation

References

Ludovic Apvrille, "Automated System Engineering with Artificial Intelligence", **Keynote** at The ESM'2023 (The 37th annual European Simulation and Modelling Conference), Toulouse, France, Oct. 24-26, 2023.

L. Apvrille, B. Sultan, "System Architects are not alone Anymore: Automatic System Modeling with AI", Proceedings of the 12th international conference on Model-Based Software and Systems Engineering (Modelsward'2024), Rome, Italy, Feb. 21-23, 2024. **Best paper award**.

L. Apvrille, B. Sultan, "AI-Driven Consistency of SysML Diagrams", accepted at **Models'2024** (Sept. 2024)

Questions?

Latest version of TTool includes what has been demonstrated. . . and much more!
`ttool.telecom-paris.fr`

All evaluation results (including models) are available here:
`https://github.com/zebradile/ttool-ai`