Together is Better! Integrating BDI and RL Agents for Safe Learning and Effective Collaboration

Manuel Parmiggiani, Angelo Ferrando and Viviana Mascardi

February 25th, 2025

Introduction and Motivation

Innovative combination of symbolic – i.e., Belief-Desire-Intention (BDI) – and sub-symbolic – i.e., Reinforcement Learning (RL) – agents.

Innovative combination of symbolic – i.e., Belief-Desire-Intention (BDI) – and sub-symbolic – i.e., Reinforcement Learning (RL) – agents.

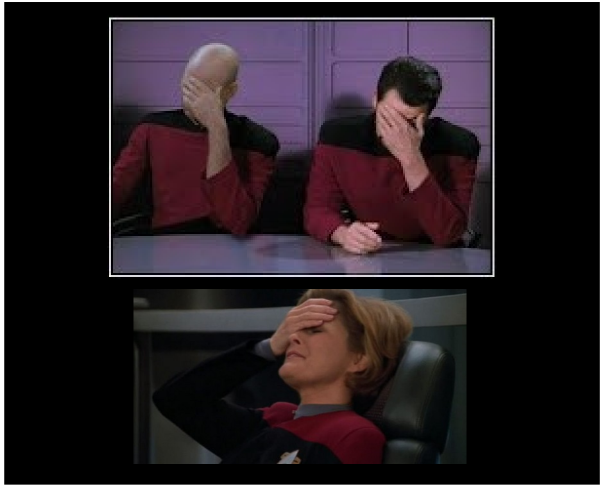
Code of **Together Is Better!** implemented in Jason[*] available at https://github.com/MAS-UniGe/TogetherIsBetter/

[*]https://jason-lang.github.io/

Innovative combination of symbolic – i.e., Belief-Desire-Intention (BDI) – and sub-symbolic – i.e., Reinforcement Learning (RL) – agents.

Code of **Together Is Better!** implemented in Jason* available at https://github.com/MAS-UniGe/TogetherIsBetter/

(and yes, the code **is** given…!)



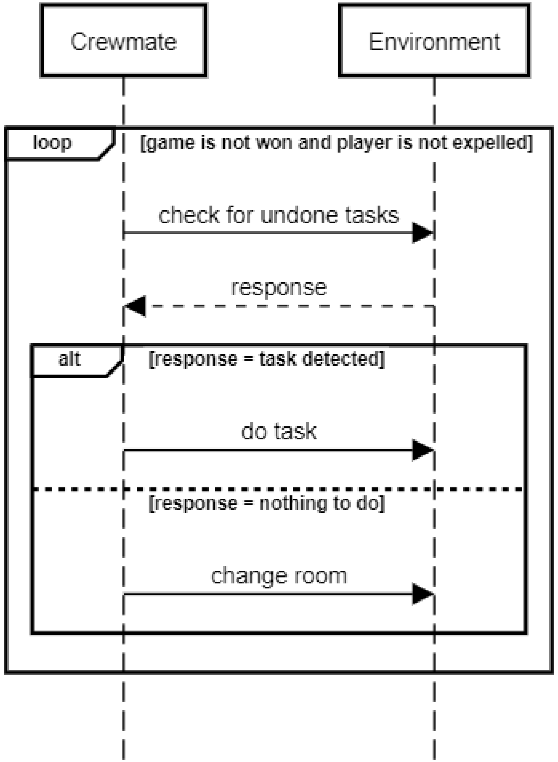
*https://jason-lang.github.io/

Case study

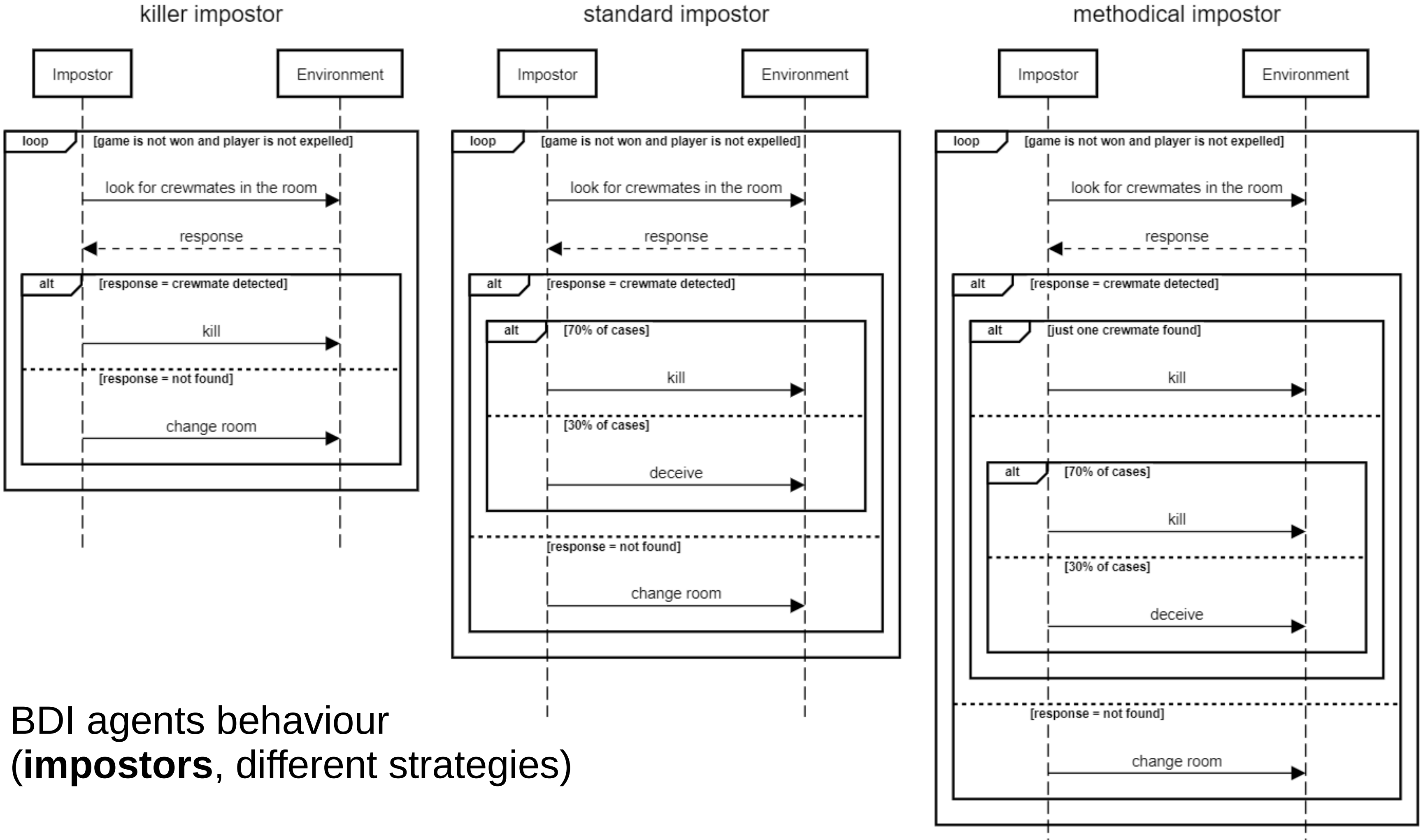

AMONG US


TOTAL TASKS COMPLETED
Ping: 50 ms
Tasks
Hannako
Tallclass
Meanfeet
Blaustoise
Saltycar
Chris A
ENGINE
ELECTRICAL
60
PROTECT
REPORT
USE

BDI agents



BDI agents behaviour
(**crewmates**, single strategy; crewmates have a skill level between 0 and 1)



BDI agents behaviour
(**impostors**, different strategies)

BDI agents implementation

First batch of experiments aimed at ***understanding which impostor strategy and which crewmate skill level would make the game as balanced as possible.***

BDI agents implementation

First batch of experiments aimed at ***understanding which impostor strategy and which crewmate skill level would make the game as balanced as possible.***

- methodical impostor selected as the optimal strategy;
- crewmate skill level of approximately 0.52 yields a win rate close to 50% for the crewmates against mthodical impostors

BDI agents implementation

First batch of experiments aimed at ***understanding which impostor strategy and which crewmate skill level would make the game as balanced as possible.***

- methodical impostor selected as the optimal strategy;
- crewmate skill level of approximately 0.52 yields a win rate close to 50% for the crewmates against mthodical impostors

We wanted to find balanced behaviours, for the **second** batch of experiments: ***having fixed (balanced) BDI crewmates, does the impostor team benefit from mixing BDI and RL agents?***

RL agents

RL agents implementation

```
learning_rate(0.2).
discount_factor(0.1).
epsilon(0.3).

@startPlan[atomic]
+start <-
          .print("starting");
          +myState(standing,noSubState).

@myStatePlan[atomic]
+myState(S1,S2) : epsilon(EPSILON) <-
          .findall(action_value(V,A),state_action(S1,S2,A,V),L);
          .random(R);
          if (R < EPSILON) {
                    .max(L,action_value(Value,Action));
          } else {
                    .shuffle(L,L1);
                    .nth(0,L1,action_value(Value,Action));
          }
          +myAction(Action);
          executeAction(Action,S1,S2).
```

Standard Q-Learning algorithm implemented in Jason

Integration of BDI and RL - Agent level

Integration of symbolic and sub-symbolic techniques at agent level

The implementation of the safe RL approach based on the teacher *[García J. and Fernández F., 2015 A comprehensive survey on safe reinforcement learning. J. Mach. Learn. Res., 16:1437–1480]* in our RL agents only requires the addition of a few lines of code to the RL agent implemented in Jason and gives the advantage of full transparency of the process, explainability, and traceability of the information learnt from the teacher.

Integration of symbolic and sub-symbolic techniques at agent level

```
!teach([s(found2orMore, kill, 3), s(found1, kill, -3)]).

@teachingPlan[atomic]
+!teach([s(State, Action, Value)|Tail]) : true
<- .send(impostor_RL_, tell,
        given_state_action(State,_,Action,Value));
!teach(Tail).

+!teach([]).
```

Integration of BDI and RL - MAS level

Integration of symbolic and sub-symbolic techniques at MAS level

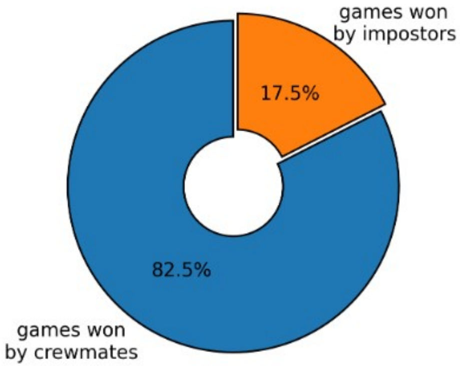


Figure 8: Win ratio when the impostor team consists **solely of RL impostors** and BDI crewmates have skill level **0.52**.

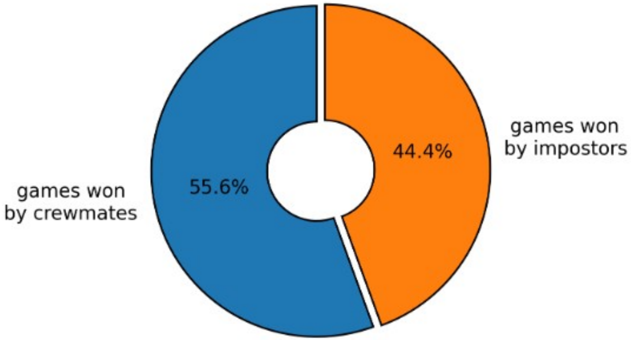


Figure 9: Win ratio when the impostor team consists of **2 RL and 2 BDI methodical impostors** and BDI crewmates have skill level **0.52**.

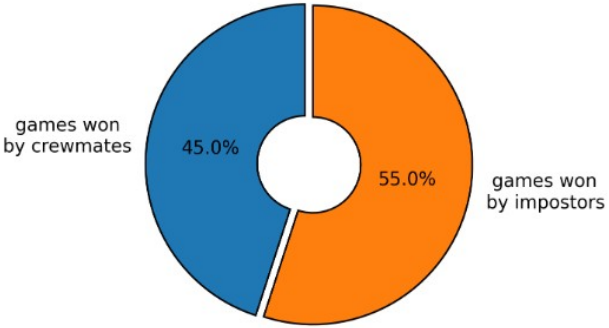


Figure 10: Win ratio when the impostor team consists of **1 RL impostor and 3 BDI methodical impostors** and BDI crew-mates have skill level **0.52**.

Integration of symbolic and sub-symbolic techniques at MAS level

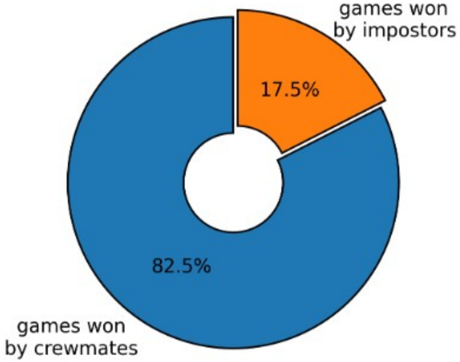


Figure 8: Win ratio when the impostor team consists **solely of RL impostors** and BDI crewmates have skill level **0.52**.

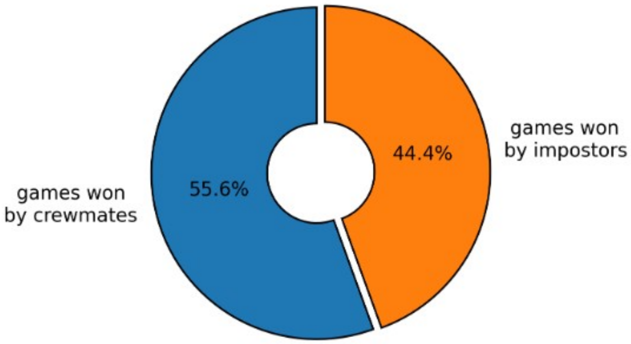


Figure 9: Win ratio when the impostor team consists of **2 RL and 2 BDI methodical impostors** and BDI crewmates have skill level **0.52**.

BDI impostor agents seem to disrupt crewmate clustering, prompting RL impostors to initiate kills earlier, resulting in a more effective team dynamics.

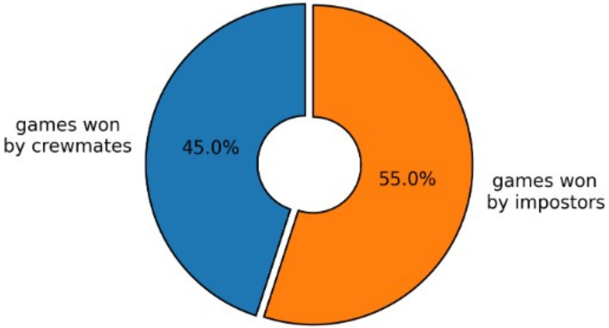


Figure 10: Win ratio when the impostor team consists of **1 RL impostor and 3 BDI methodical impostors** and BDI crewmates have skill level **0.52**.

Comparison

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
→ SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
→ SafeRLJ	No	Yes	No	Yes	Compl.	(Not given, 2024)	MIT

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
SafeRLJ	No	Yes	No	Yes	Compl.	(Not given, 2024)	MIT

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
SafeRLJ	No	Yes	No	Yes	Compl.	(Not given, 2024)	MIT

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
→ SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
→ SafeRLJ	No	Yes	No	Yes	Compl.	(Not given, 2024)	MIT

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
→ SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
→ SafeRLJ	No	Yes	No	Yes	Compl.	~~(Not given, 2024)~~	MIT

The code of **Together Is Better!** implemented in Jason is available at https://github.com/MAS-UniGe/TogetherIsBetter/

Table 1: General information on recent proposals for integrating BDI and RL agents.

Ref.	Year	Language	RL Algorithm	Implementation of RL
(Badica et al., 2015)	2015	Jason 1.4.2	TDL	Jason Plan
(Badica et al., 2017)	2017	Jason 1.4.2	SARSA	Jason Plan
(Wan et al., 2018)	2018	Jason	SARSA	Jason Plan
(Bosello and Ricci, 2019)	2019	JaCaMo 0.7	SARSA	Jason Action
(Pulawski et al., 2021)	2021	Jason-RL	Iterative SARSA	As in (Bosello and Ricci, 2019)
(Persiani and Hellström, 2022)	2022	Python	DeepRL predict.	DeepRL in Keras
(Ichida and Meneguzzi, 2023)	2023	Python AS	Q-Learning FA	Outside
→ SafeRLJ	2024	Jason 3.1	SARSA	Jason Plan

Table 2: Features, examples, and licensing of recent proposals for integrating BDI and RL agents.

Ref.	BDI Integration	Safe	Hybrid Planning	Hybrid MAS	Example	Code	License
(Badica et al., 2015)	No	No	No	No	Toy	No	
(Badica et al., 2017)	No	No	No	No	Toy	(Felton, 2017)	GPL-3.0
(Wan et al., 2018)	No	No	Yes	No	Toy	No	
(Bosello and Ricci, 2019)	Yes	No	No	No	Compl.	(Bosello, 2019)	LGPL-3.0
(Pulawski et al., 2021)	As in (Bosello and Ricci, 2019)	No	No	No	Toy	No	
(Persiani and Hellström, 2022)	No	No	No	No	Toy	(Persiani, 2022)	MIT
(Ichida and Meneguzzi, 2023)	Yes	No	Yes	No	Compl.	No	
→ SafeRLJ	No	Yes	No	Yes	Compl.	~~(Not given, 2024)~~	MIT

(and yes, the code **is** given…!)

Thank you for your attention!

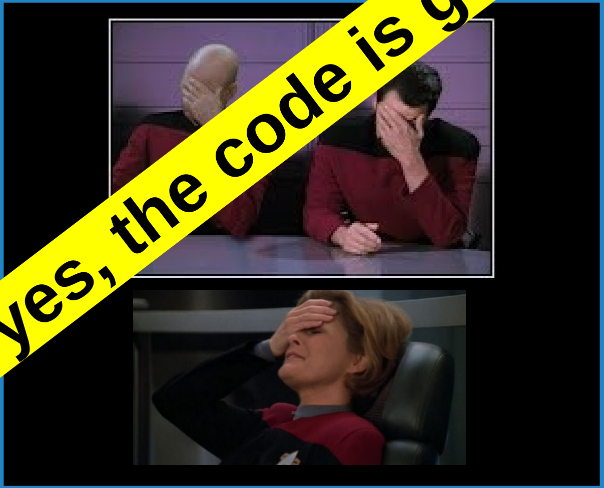
Any questions?

You can find us at
manuel.parmi@gmail.com
angelo.ferrando@unimore.it
viviana.mascardi@unige.it

Thank you for your attention!

Any questions?

You can find us at
manuel.parmi@gmail.com
angelo.ferrando@unimore.it
viviana.mascardi@unige.it



and yes, the code is given ;)