



Multi agents simulation of justice trials to support control management and reduction of civil trials duration

Beniamino Di Martino^{1,2,3} · Antonio Esposito^{1,3} · Luigi Colucci Cante^{1,3}

Received: 12 May 2021 / Accepted: 30 August 2021 / Published online: 18 September 2021
© The Author(s) 2021

Abstract

Thanks to the digitalization of Civil Legal Processes that has been carried out in Italy since 2005, a consistent flow of data has been generated regarding all phases of Civil Trials. The Italian Ministry of Justice has recognized the opportunity to better assess the quality of Courts' management, seeking ways to implement Decision Support Systems to aid the work of Court Presidents. In collaboration with the Court of Livorno, which has provided data and case studies, KPIs and indicators have been developed, and have been later used within an Agent-based simulation framework to assess the behaviour of Chancellors and Judges in response to Court Presidents' decisions, and in particular to verify the effects of such decision on Processes' duration. This paper presents the Agent-based simulation approaches adopted to predict the effects of Presidents' decisions, by taking in consideration the behaviour of Chancellors and Judges, derived from the examination of past Trials, whose aspects have been recorded by the Court's digital system.

Keywords Simulation · Multi-agent · JADE · Management control

1 Introduction

Computerization and automation have experienced a huge leap forward in the last decade, and the progressive digitalization has brought a data deluge that resulted in the Big Data phenomenon. Companies and corporations are gaining benefits from this increased volume of available data, which has become fundamental to the extraction of knowledge that is at the base of modern decision systems. Big data, process mining and machine learning techniques have been employed to design and extract accurate models of reality, that are then used to try to better understand the working environment and

perform informed decisions. Simulation, as one of the main techniques used in decision making to assess the outcomes of taken actions, has been adopted in several situations, from Building Information Modeling to e-health, to predict the behaviour of modelled systems.

Italian Courts have also collected a consistent quantity of data in their databases, through which the Ministry of Justices has been trying to implement frameworks that, according to predefined management policies, can provide Courts' Presidents the means to better control their Courts. The available data regard every aspect of a Court, allowing the definition and calculation of complex key performance indicators (KPIs) that can even analyze the amount of work done by each employee in the Court and the time that was necessary to perform each single task (Di Martino et al. 2021e). Making decisions related to the allocation and redistribution of resources is thus possible.

Of course, since the quantity of available data is outstanding, and their structure is not always well defined (written documents are directly uploaded to the Court's database, without previous analysis), Big Data analysis tools are needed, and ad-hoc pipelines need to be implemented (Di Martino et al. 2021d, b, c).

This paper proposes a new methodology to implement a management control policy in an Italian Court through

✉ Beniamino Di Martino
beniamino.dimartino@unicampania.it

Antonio Esposito
antonio.esposito@unicampania.it

Luigi Colucci Cante
luigi.coluccicante@unicampania.it

¹ Department of Engineering, University of Campania "Luigi Vanvitelli", Via Roma 29, 81031 Aversa, Italy

² Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan

³ CINI-Consortio Interuniversitario Nazionale per l'Informatica, Rome, Italy

Multi-Agent simulation techniques. This methodology is applied to a specific dataset provided by the **Court of Livorno** (*Tribunale di Livorno*), to validate it through a concrete Case Study. The data have been collected through **SICID** (Sistema Informatico Contenzioso Civile Distrettuale), that is one of the information systems that ensure the functionality of the PCT (**Online Civil Trial** (*Processo Civile Telematico*)). SICID is accessible from all Chancelleries of the Italian judicial offices, and is used to manage and collect data for all possible Civil Trials, such as litigation, labor disputes, and insolvency procedures (Lupi 2017).

The work presented in this paper stems from previous research efforts that aimed at the definition of KPIs and indicators (Di Martino et al. 2021a), which are necessary to correctly assess the overall performances of a Court and its employee. Such KPIs have been realized according to standard definitions used within the European Union, and have been further refined to meet the expectations of Court Presidents. Furthermore, a dual Big Data pipeline for the analysis of both structured data, resident in the Court's databases, and unstructured data (sentences and documents which do not have a structured format) has been developed for the calculation of such KPIs (Di Martino et al. 2021d). These works can be considered preliminary and necessary to implement the simulation system that is described here. The remainder of this paper is organized as follows: Sect. 2 shows an overview on some similar works to content of this paper and introduces some background concepts as multi-agent systems (MAS), intelligent agents, multi-agent simulation's techniques, and JADE platform; Sect. 3 describes main concepts of management control applied to the Court of Livorno; Sect. 4 introduces the ad-hoc methodology developed to perform the multi-agent simulation, and Sect. 5 describes the key performance indicators (KPIs) used to evaluate performance of chancellors, judges and trials in multi-agent simulation. Section 6 describes a prototype tool that implements this methodology, and all the results taken out of the simulation. The paper ends with Sect. 7 with conclusions and future works directions.

2 Related work and background

The use of multi-agent system based techniques has grown exponentially in recent years. As a result, many research works have been carried out towards the definition of multi-agent platforms and frameworks to simulate time series of data. In particular, we are interested in works where agents are used to make predictions on future events and can simulate “what-if” scenarios. The work published in Stranjak et al. (2008) proposes a simulation tool, based on the Jade platform, that is used to simulate the behaviour of aero engines and, in particular, to schedule their overhaul time.

The objective is to reduce the “out-of-order” time of engines by efficiently schedule their maintenance.

Human behaviour and the consequences of human choices are instead the focus of the work presented in Yue et al. (2020). The authors describe a simulation model used to predict the effects of different government policies on human choices regarding the consumption of energy in different towns in China, with the objective to increase energy saving. The work underlines the importance of simulation, supported by Neural Networks.

Another interesting work involving human behaviour simulation for the prediction of choices' outcomes is reported in He et al. (2021). The work describes the first open-source multi-agent simulation model for New York City, called MATSim-NYC, which supports the simulation of traffic congestion in New York, trying to identify how people react to congestion policies and to mitigate traffic problems.

The problem of task scheduling, which is central in our work as we want to re-schedule Chancellors' and Judges' activities optimally, is addressed in Zhu et al. (2019), where the authors describe a multi-agent system where a greedy optimization algorithm is applied. The authors demonstrate the efficacy of Agents in scheduling tasks and actions by applying greedy algorithms, thus showing the feasibility of the approach.

While the use of Multi Agent systems for simulation is commonly applied in several scenarios for scheduling or prediction purposes, very little has been done in the juridical domain. In Zhong et al. (2013) the authors use a multi agent based system to simulate a Court trial, but their approach is limited to a single case study and to a very limited number of actors. The work presented in Mishra and Singh (2017) extends the Gaia MAS methodology (Wooldridge et al. 2000) with a Goal Model, which has been then tested against a Court use case, related in particular to the “Legal Semantic Web”.

While these works address the possibility to simulate Court trials, they do not pursue specific goals in the direction of optimizing the trial management system, or to increase the overall performances of Courts, as they are more interested in showing the general efficacy of their models and methodologies.

The work presented in this paper differs from the existing approaches, as it directly applies simulation techniques based on multi-agent systems (MAS) (Logenthiran et al. 2010) to real data provided by an Italian Court, in order to efficiently reschedule the Court's activities and reduce the duration of Civil Trials. This is an important topic that, as far the authors of this manuscript know, has not been addressed anywhere else, at least not through the use of Agents.

The use of Multi-Agent technologies was of vital importance in this paper, as they made it possible to carry out an agent based simulation (ABS) in which the social behaviour

of an entire court and all the interactions between the various players in the court were reproduced, and the distribution of workloads in the organisational units was simulated.

The JADE agent platform was used to implement the multi-agent system. JADE (Java Agent DEvelopment Framework) is a framework that supports the development of distributed applications based on the Agent programming paradigm. JADE was conceived and developed at *TILAB (Telecom Italia Lab)* in Turin, the Research and Development branch of the *Telecom Italia group*, and is an open source software that is distributed under LGPL license, as reported in Simone et al. (2013).

3 Management control applied to the Court of Livorno

Management control is an attempt to implement, in an organized and systematic way, a planning and monitoring process considered essential in any complex structure (private company or public office) to manage resources (human, instrumental and financial) in a rational way (Massimo Orlando 2000). The Court of Livorno is implementing a policy for management control, in the hope that through a combination of process mining and multi-agent simulation techniques, a tool can be provided to the President of the Court that allows him to analyze the amount of work done by each employee, the time required by each task and, consequently, helps in making decisions related to the allocation and redistribution of resources. The key points for which the Court of Livorno adopted this orientation are briefly summarized:

- The recording activities are managed almost autonomously by Chancellors and, in many cases, they escape the control of the President of the Court. Therefore, it was deemed necessary to proceed with the definition of procedures to acquire useful information regarding all Chancellery services. So it was decided to proceed with their analysis to identify any critical issues and develop possible solutions to be included in the plans.
- Administrative activities also need to be monitored more carefully in order to recognize and correct potential issues.
- It is indispensable to monitor the work of Judges and Chancellors and to correct misbehaviour. On the one hand each Judge, aware of the fact that the President of the Court will periodically examine the progress of his cases, will be motivated to pursue productivity in line with that of other colleagues. On the other hand also the Chancellor manager, knowing that at the end of the year (or quarter or semester) he will have to provide reports regarding the specific activities attributed to his office,

will also try to avoid the accumulation of the office backlog.

- The main aim is learning from past history (management data) to understand where to intervene to improve the judicial service and the quality of the work of Judges and administrative staff.
- The management control contributes to facilitate the implementation of regulatory changes and/or the organizational changes necessary to innovate and improve the overall performance of a judicial office.

Definitely, the management control project attempts to provide the judicial office with a tool for collecting management data and that is able to identify the points of excellence of the organizational structure, but also its flaws: in this way it's possible to prevent criticalities or to facilitate their removal.

A rich database has been made available by the Court of Livorno to perform these analysis: it contains a list of all the Civil Trials managed by the Court of Livorno over the past 20 years, together with the history of all the events that affected such Trials. Table 1 summarizes data provided by the Court of Livorno.

4 Methodology

This section describes the ad-hoc methodology developed to perform the multi-agent simulation. For the sake of clarity, such a methodology has been divided into sub-phases, as also shown in the workflow reported in Fig. 1.

These substeps are necessary to prepare the data collected from the Court's database, and create the behavioural model employed by Agents.

1. Data modelling, preparation and ingestion: a model for the data contained in the historical archive has been developed, to correctly ingest them, reduce redundancies and identify incoherence;

Table 1 Data provided by the Court of Livorno

Data	Value
Number of processes	183,810
Number of records in the archive	2,444,014
Number of matters	41
Number of objects	143
Number of possible events	774
Number of possible states	170

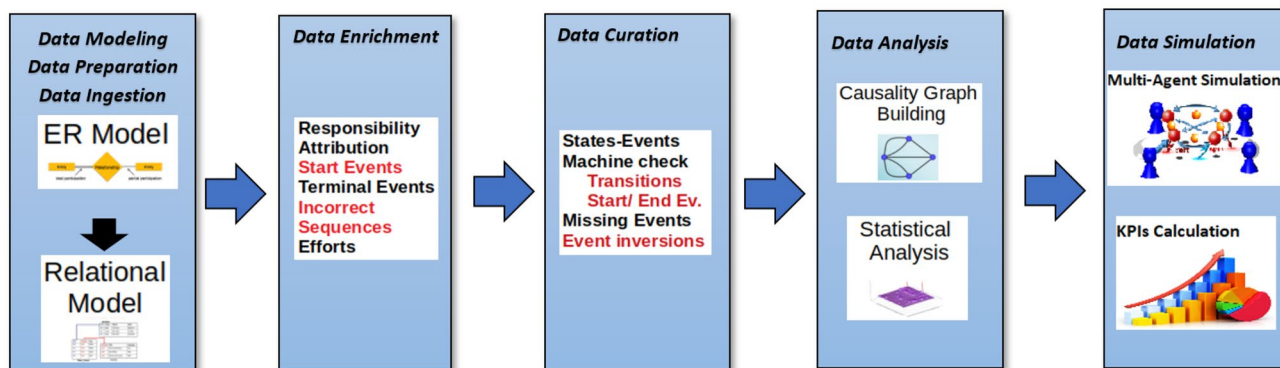


Fig. 1 Phases of the proposed methodology

2. Data enrichment: the base data model has been extended, while the data extracted from the archive has been integrated with information from domain experts;
3. Data curation: once all data has been correctly imported and additional information has been harvested, it is possible to check it to identify contradictions and incoherence, and act to solve them;
4. Data analysis and process model extraction: once all the data has been gathered and incorrect information has been removed, it is possible to actually mine the process model and to represent it for further analysis; here causality graphs are built;
5. Data simulation: once the model is ready, it is possible to proceed with the event simulation starting from the causality graph produced previously. Finally different simulation scenarios can be proposed. The aim of multi-agent simulation is to simulate the past story, in order to use the knowledge learned from the past to optimize the process model. The past history can be slightly modified to verify if some assumptions are correct.

The first three substeps have been actually implemented in Di Martino et al. (2021e), while here we are focusing on the following phases.

4.1 Causality graphs' building

Building causality graphs is necessary to perform the multi-agent simulation.

A causality graph is a representation of the causal sequences of events in terms of a Graph, in which the vertices represent the events of the judicial proceeding, and the edges represent transitions from a previous event to the next one, within the same judicial Trial.

A causality graph contains a number of events, for each of which a set of attributes is stored:

- The actor responsible for the event.

- The event's description.
- The alphanumeric code identifying the event.
- The total occurrence number of the event in all traces of the historical archive.
- If the event can be considered as terminal for Chancellors (TC) or Judges (TG).
- The average recording time required by Chancellors to perform its registration.

For each occurrence of an event in every trace, other information are recorded in the graph structure:

- The trace ID.
- The ID of the Judicial object the trace refers to.
- The dates of occurrence (Dataev) and of registration (Datare) of the event.
- The position (as a step number) of the event in the trace.
- The ID of the Chancellor who made the registration.
- The ID of Judge defining the process.

The graph also contains a number of edges, for each of which a set of attributes is stored:

- The source and target events of the transition.
- The total occurrence number in all traces of historical archive.
- The average time and average variance to perform a transition between two events.

For each occurrence of an edge in every trace, some information are recorded in this structure, such as the trace's ID, the position of the transition in the trace, and the transition's duration in days.

A reduced version of Causality Graph of matter 453 (SOCIETARIO CAMERALE PRIMO GRADO) is shown as an example in Fig. 2.

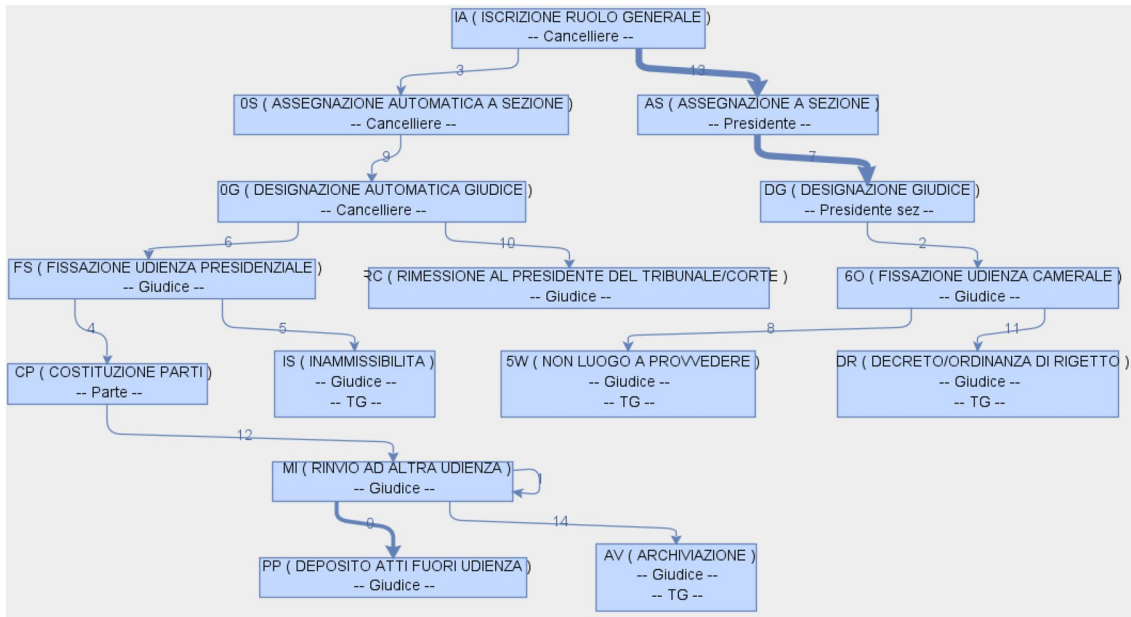


Fig. 2 Causality Graph of Matter 453

4.2 Simulation workflow

The last phase of the methodology described in Sect. 4 proposes the construction of a Multi-Agent Model that simulates the behaviour of each Actor in the Trials. Once the model is ready, we can proceed with the event simulation from the causality graph produced previously. Figure 3 shows the Simulation Workflow: the Event Simulator’s block get in input (i) all Causality Graphs produced for each subject of the historical archive, and (ii) the estimation of the time required by actors to complete an event (effort).

The simulation output consists of a distribution of log concerning the recording activities performed by each Chancellor in the simulation. Every record in this log file contains some information as example the code of Chancellor that has perform the registration, the day on which he registered, the organisational unit in which the clerk was working that day, and the number of days it took to complete the registration. A KPIs calculation module read these log files, and produces some statistics and plots that support Judges in making informed decisions.

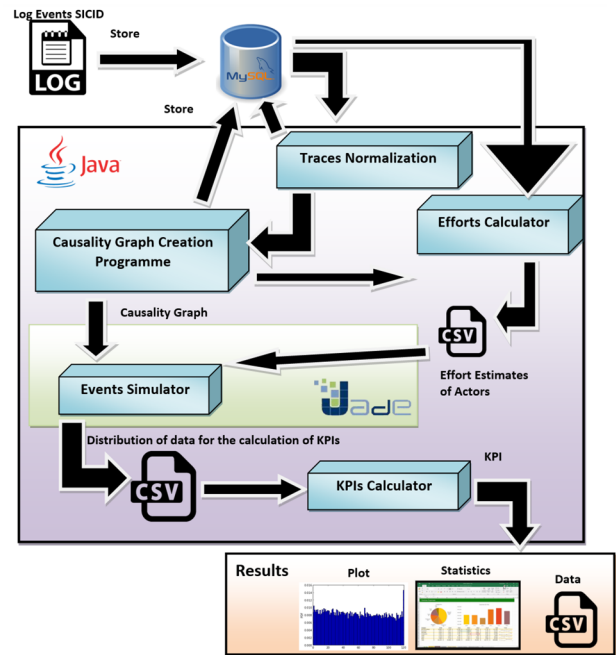


Fig. 3 Simulation workflow

5 KPIs

This section provides a short introduction to the KPIs developed for the realization of the simulation tool. A more precise description of such KPIs is reported in Di Martino et al. (2021a). From the simulation of past history, through the calculation of these KPIs, certain slowdowns,

bottlenecks, and outliers become evident. In this way, the system highlights the points where there have been major problems, on which the court president must intervene by making appropriate decisions (e.g. by increasing the number of chancellors on certain processes).

In particular, three kinds of KPIs are calculated by the simulation tool: KPIs related to Chancellors, Organizational Units, and whole Trials. Concerning to Chancellors the first KPI calculated is the average recording time for a Chancellor, that is defined as shown in Eq. 1.

$$TM = \frac{\sum_{i=1}^N DR_i - DE_i}{N}. \quad (1)$$

This KPI is defined as the average time lapse between the date a registration request is sent to a Chancellor (Date of any Event generated by any actor other than the Chancellor itself), and the Date on which the Chancellor actually records the event. The numerator of this ratio is the difference in days between the Date of Recording of the event by the Chancellor DR_i , and the Date of the event DE_i . All of these time differences must be added together, and the result must be normalized by dividing by the total number of registration events N made by that specific Chancellor.

Another KPIs calculated concerning to Chancellors is the *Productivity*, defined as the number of recording activities carried out by a specific Chancellor in the simulation period.

Concerning to Organizational Units (UO) the KPI calculated are the average recording time, productivity and working capacity. The *Working Capacity* is an indicator of the work performed by each Chancellor in the actual Organizational Unit defined in equation 2:

$$CL = \sum_{i=1}^K \frac{N_k}{Act}. \quad (2)$$

This KPI is calculated by evaluating, day by day, the weight of the work done by each Chancellor, using the total number of registration activities performed in that day for comparison. In particular, the contribution of the Chancellor who worked the most is set at 1, while the contribution of other Chancellors is calculated as a fraction between 0 and 1. The following parameters were used in the Eq. 2: *Act* is the total number of recordings made by the most operational Chancellor; K is the number of Chancellors involved in the UO; N_k is the number of events recorded by a specific Chancellor.

For each Unit it is possible to calculate an *average registration time at organizational unit level* whose definition is given in the Eq. 3:

$$TM = \frac{\sum_{j=1}^M DR_j - DE_j}{M = \sum_{i=1}^K N_k}. \quad (3)$$

The following parameters were used in the Eq. 3: M is the total number of events recorded; K is the number of Chancellors involved in the UO; N_k is the number of events recorded by a specific Chancellor.

Calculations also include the average recording time normalized with working capacity or with productivity, or the productivity normalized with working capacity.

Concerning to trial duration, three KPIs are calculated: maximum duration of real and simulated trials; average duration of real and simulated trials; number of real and simulated trials that exceed X days, where X is a value indicated by the user.

6 Realization

A prototype tool that implements the methodology described in Sect. 4 has been developed. This tool was developed entirely in Java, and exploiting the JADE platform described in Bellifemine et al. (1999, 2000, 2002) that was used to implement the multi-Agent model.

The graphical tool interface is showed in Fig. 4.

The graphical tool interface is composed by three main panels: there is a panel to choose configuration settings, to start, stop or pause the simulation, and to display a simple report of the data to be simulated, containing the number of actors in the simulation, the number of processes and events involved. A second panel shows all the messages exchanged among the various actors during the simulation, using different colours for each type of actor. Have been used the following colours:

- Teal for Chancellor's messages;
- Red for Judges' messages;
- Dark green for Parties' messages;
- Dark orange for the Court President's messages;
- Brown for the section Presidents' messages;
- Magenta for other actors' messages (CTUs, lawyers, etc.);

Finally a third panel shows, for each simulated working day, the effort and workload queue. The workload queue contains, for each actor in the simulation, the number of events that that actor performs on a certain date. The effort queue contains, for each actor in the simulation, the working time, expressed in minutes, required for that actor to complete all the activities he is in charge of on a given date.

The functionalities provided from the events tab are shown from the UML use case diagram in Fig. 5.

Once the final user has chosen the period of analysis, the execution speed and a simulation scenario, it is possible to start the simulator. In reality, not all days in the chosen time frame will be simulated, but a small algorithm has been developed that searches only for actual working days in which at least one actor has performed an action. At any time during the simulation it is possible to pause the simulation, resume it, or even stop it in advance before the

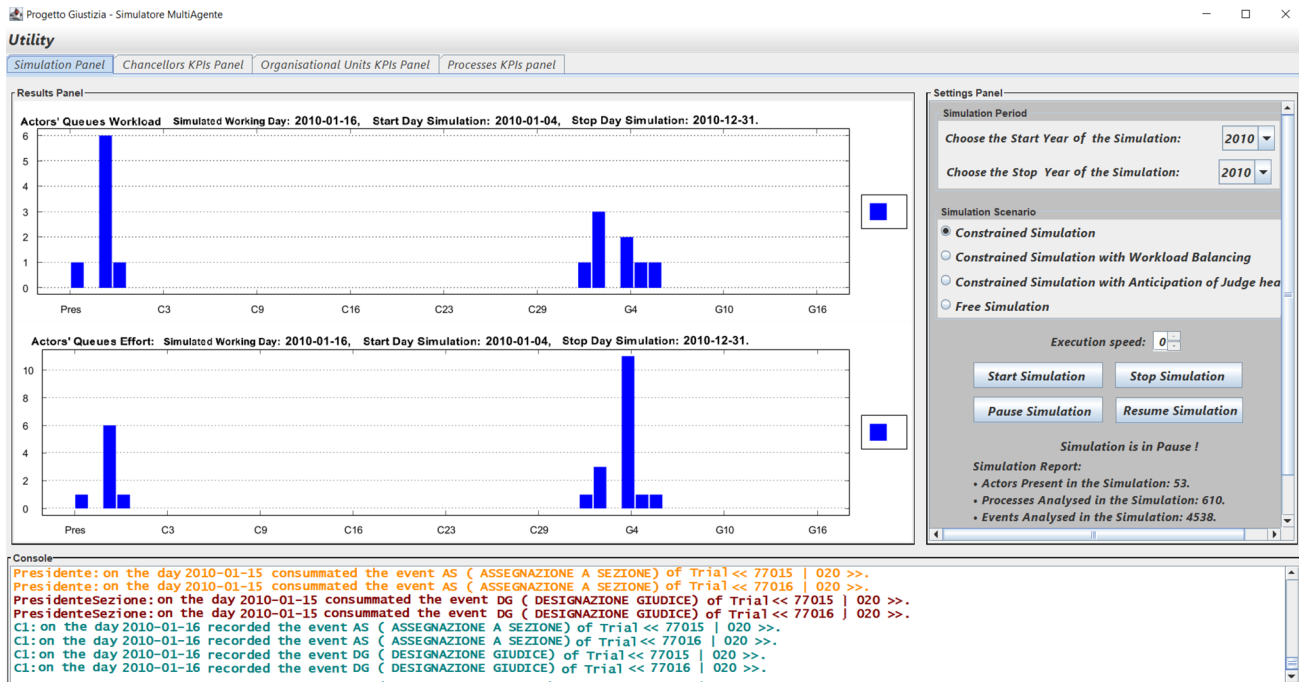
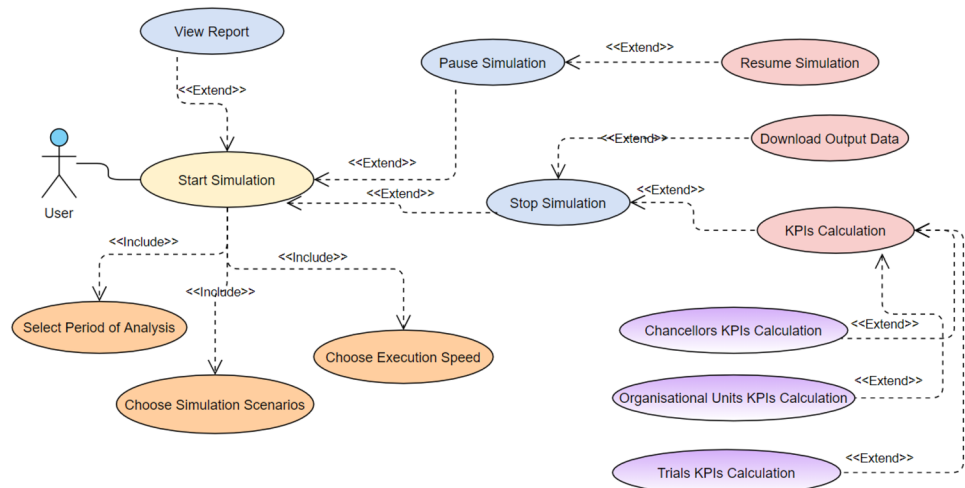


Fig. 4 Graphical tool interface

Fig. 5 Multi-agent simulator use case diagram



final simulation date has arrived. As soon as the simulation finishes or is stopped, a KPIs calculation module is launched to calculate the KPIs concerning the Chancellors, the organisational units, and the Trials.

This tool provides the possibility of using the JADE's utility as a Sniffer, to keep track of messages exchanged between all actors. Figure 6 shows an example of the Sniffer in use to simulate a single Trial.

As shown in Fig. 6, two kind of JADE's messages are used: (i) a *REQUEST* message to indicate that the event has been consumed, and the actor who was responsible for

it sends the next message to be executed to the right actor; (ii) an *INFORM* message to specify requests for registration of an event with the responsible Chancellor.

Three simulation scenarios have been defined in this paper:

- Constrained simulation;
- Constrained simulation with workload balancing;
- Constrained simulation with anticipation of Judge hearing events.

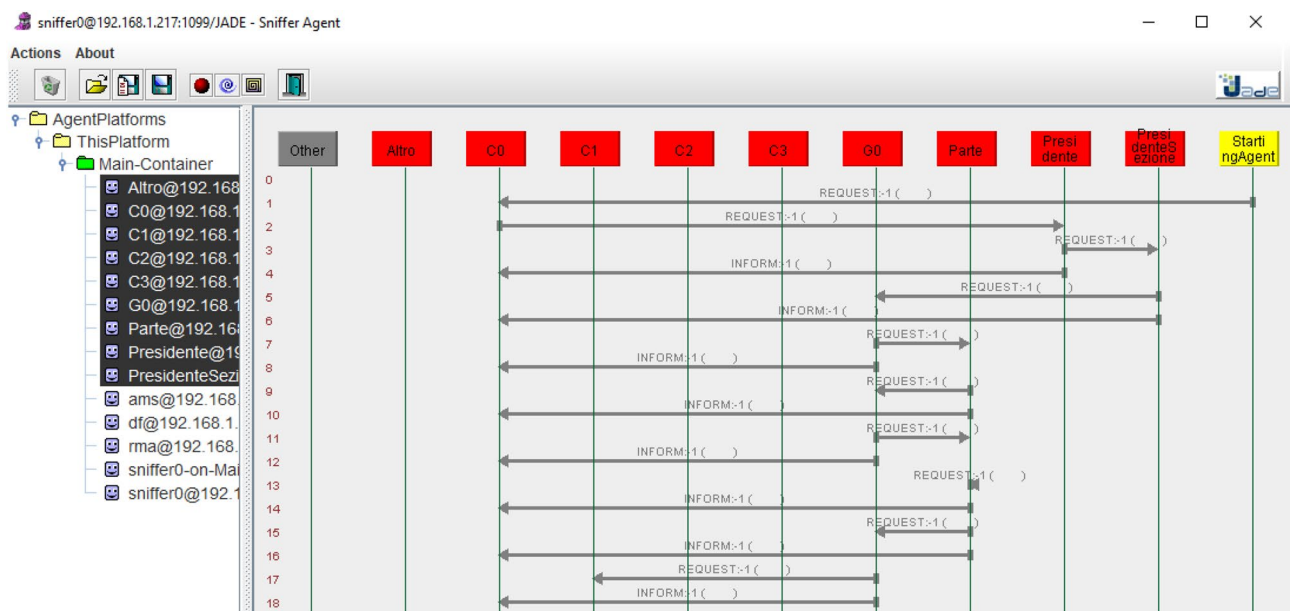


Fig. 6 Example of sniffing of messages exchanged in JADE Platform

Constrained simulation This is the most constrained scenario, because it involves reconstructing reality as it happened. Events are assigned to the exact same actors who performed them in reality on the exact same dates.

The algorithm that implements this simulation scenario is described in more details below:

- StartingAgent takes care of the creation of all other Agents involved in the simulation: a single Agent is created for the actors “Part”, “President”, “PresidentOfSection”, and “Other”, while different instances of the actors “Chancellor” and “Judge” are created.
- The StartingAgent creates a Calendar Data structure C, whose starting date d0 is the smallest of all process dates in the simulation.
- As soon as all the Agents have been created, the StartingAgent starts a clock to simulate the flow of the Court’s working days. This clock has been implemented as a JADE TickerBehaviour: every XX seconds the StartingAgent generates a clock stroke, and if all Agents have finished their message exchanges during that working day, then it increments the calendar by one working day.
- If there are initial events of a Trial (i.e. of type IA-ISRIZIONE RUOLO GENERALE) whose date of occurrence (DATAEV) is equal to the simulation date, the StartingAgent can start that process by sending the initial message to the respective Chancellor that started it in reality.
- Each actor Agent A.i implements a JADE CyclicBehaviour “StoreEvent” which puts the latter in continuous waiting for messages: If the received message is of type

REQUEST, then it is received by A.i, in all other cases it is ignored; A.i extracts the event E from the received REQUEST message and stores E in its own queue; A.i updates its workload and effort in the shared Data structure.

- Each actor Agent A.i implements a JADE CyclicBehaviour “ConsumeEvent” that forces the latter to perform a continuous polling on the Calendar C.d. and if A.i. in its queue contains an event E such that its DATEEV is equal to the simulation date in the calendar, then:
 - (a) Event E is consumed and removed from the queue;
 - (b) A.i updates its effort and workload from the shared Data structure;
 - (c) A.i sends a message of type INFORM to the Chancellor responsible for event E, containing all information on event E;
 - (d) A.i queries the causality graph to find out who is the actor responsible for the next event (E.next);
 - (e) A.i sends a REQUEST type message to the next actor, containing all information about the E.next event;
- Each Chancellor CA.i implements a JADE CyclicBehaviour “RecordEvent” which puts the latter in continuous wait to receive messages of type INFORM. When it receives one:
 - (a) CA.i extracts the event E from the message INFORM;
 - (b) CA.i stores E in its queue;

- (c) CA.i updates its effort and workload from the shared Data structure;
- Each Chancellor CA.i implements a JADE CyclicBehaviour “ConsumeEventRegistration” that forces the latter to perform a continuous polling on the Calendar C.d. and if CA.i. in its queue contains an event E such that its registration date (DATARE) is equal to the simulation date in the calendar, then:
 - (a) E is consumed and removed by CA.i’s queue;
 - (b) CA.i updates its effort and workload from the shared Data structure;
- The StartingAgent every XX seconds reads the effort and workload of each actor A.i, and shows the data on the screen in a histogram automatically generated.

Constrained simulation with workload balancing Event efforts are no longer assigned to the same actors who performed those events in reality. The causal link between the actors is relaxed: the workload of Chancellor’s events (both registration activities and normal events) is equally balanced between the various Chancellors assigning the next task to the Chancellor who is currently the freest, thus eliminating any bottlenecks in the system. Figure 7 shows the effects of this simulation scenario on the workload and effort queues of the various actors.

Constrained simulation with anticipation of judge hearing events The temporal causality Constraint is relaxed by modifying the date on which events are considered to happen. The user is given the option of setting a threshold,

expressed as a fixed number of days: all Trials exceeding this threshold are considered slow. When a judge receives has to perform a hearing event, and the Trial to which such event belongs is considered slow, then that judge can anticipate the event by X Days, where X is a random number of days between the current date and the date on which that event would have been consumed in reality. These X days anticipated by the slow Trials will have to be taken from the various “non-slow” Trials, which at that moment have at least one event in the queue of that Judge, by distributing the lost time equally.

The Key Performance Indicators computed for the various simulation scenarios are analysed in the following. The entire year 2010 was chosen as the simulation period, so all Trials whose initial IA (ISCRIZIONE A RUOLO) event occurs in 2010 are simulated. For this simulation period 610 Trials have been simulated, with a total of 4538 events and 53 actors involved (of which 32 Chancellors and 17 Judges). The Chancellors’ KPIs described in Sect. 5 and obtained in simulation scenarios “Constrained Simulation” and “Constrained Simulation with Workload Balancing” are compared in Fig. 8.

In particular Fig. 8b emphasises that by balancing the workload between the various Chancellors, their Productivity fluctuates around the average value of 139 events registered in total, during the period. In addition, the average recording time of Chancellors is reduced from 2.17 events per day to 2.02. These results show that, just by changing the scheduling of events, it is possible to reduce the average registration time of events. The UO’ KPIs described in Sect. 5 and obtained in simulation scenarios

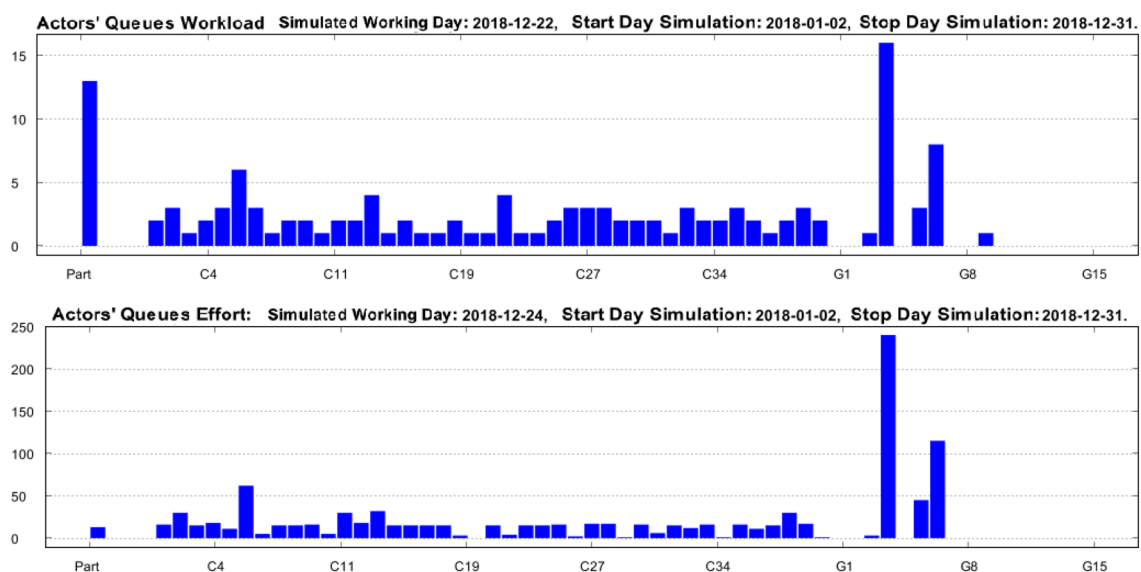
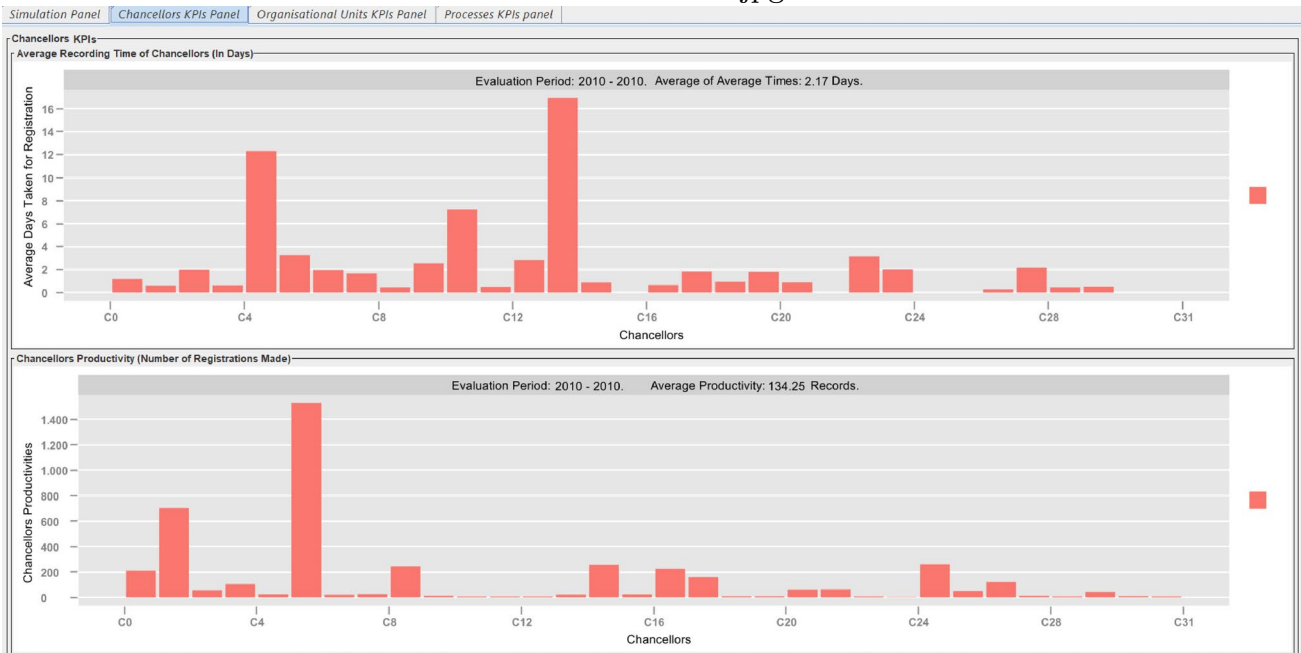


Fig. 7 Simulation with workload balancing

2010 cancellieri.jpg



(a) Chancellors' KPIs in Constrained Simulation.
 2010 cancellieri.jpg



(b) Chancellors' KPIs in Workload Balancing Simulation.

Fig. 8 Chancellors' KPIs

2010 UO.jpg



(a) UO' KPIs in Constrained Simulation.

2010 UO.jpg



(b) UO' KPIs in Workload Balancing Simulation.

Fig. 9 UO' KPIs

“Constrained Simulation” and “Constrained Simulation with Workload Balancing” are compared in Fig. 9.

A comparison of Fig. 9a, b shows that organisational units haven't improvement on average recording times by balancing workloads, but there is an increase in the average

productivity of the various units. This improvement is due to the fact that the workforce in case Fig. 9b was better distributed among the various units, in fact compared to case Fig. 9a there is an average work capacity of about 15 Chancellors per unit (against about 3 Chancellors per unit

PROCESSI 2010 oltre 200 giorni.jpg



Fig. 10 Trials' KPIs

in constrained simulation). Finally an improvement in KPIs concerning Trials is shown in Fig. 10.

A threshold of 200 days after which any process is considered slow was chosen in the simulation. This improvement was obtained in a Constrained Simulation with Anticipation of Judge hearing events scenario. In particular, Fig. 10 shows that the average duration of Trials has been reduced from 113.93 to 111.08 days, and the number of Trials exceeding 200 days has been reduced from 146 to 124.

7 Conclusions

In this paper a methodology for analyzing the past history of a Trial through a large set of management data, and understanding where to intervene to improve the quality of work of Judges and Chancellors has been provided. This methodology is based on the use of Multi-Agent simulation techniques and on the analysis of a specific set of KPIs. A prototype tool implementing this methodology has also been presented, and results have been illustrated. The use of this tool may provide a relevant support for the implementation of Management Control in Italian Courts, offering the President of the Court a valuable decision support.

In future work, the simulation could be enriched with new, more interesting simulation scenarios: for example, we are already evaluating the possibility of using Machine Learning techniques or more sophisticated statistical criteria based on the use of Markov chains to implement a scenario

of free simulation, in which each actor, once consumed an event, can choose “almost freely” to which next actor send the next event based on the reading of the causality graph. It will not be a completely free simulation in how much it will be necessary to introduce specific controls that prevent the generation of simulated processes that are not valid.

Acknowledgements The work described in this paper has been funded by the Applied Research Project “Big data Giustizia e Datawarehouse” promoted by the Italian Ministry of Justice and realized by Consorzio Interuniversitario Nazionale per l’Informatica (CINI).

Funding Open access funding provided by Università degli Studi della Campania Luigi Vanvitelli within the CRUI-CARE Agreement.

Data availability statement The data that support the findings of this study are available from the Italian Ministry of Justice but restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bellifemine F, Caire G, Trucco T, Rimassa G (2002) Jade programmers guide. *Jade Version 3*:13–39
- Bellifemine F, Poggi A, Rimassa G (1999) Jade—a fipa-compliant agent framework. In: *Proceedings of PAAM, London*, vol 99, p 33
- Bellifemine F, Poggi A, Rimassa G (2000) Developing multi-agent systems with jade. In: *International workshop on agent theories, architectures, and languages*. Springer, pp 89–103
- Di Martino B, Colucci Cante L, Esposito A, Lupi P, Orlando M (2021a) Supporting the optimization of temporal key performance indicators of Italian courts of justice with olap techniques. In: Barolli L, Yim K, Enokido T (eds) *Complex, intelligent and software intensive systems*. Springer International Publishing, Cham, pp 646–65. https://doi.org/10.1007/978-3-030-79725-6_65
- Di Martino B, Marulli F, Graziano M, Lupi P (2021b) Prettytags: an open-source tool for easy and customizable textual multilevel semantic annotations. In: Barolli L, Yim K, Enokido T (eds) *Complex, intelligent and software intensive systems*. Springer International Publishing, Cham, pp 636–645. https://doi.org/10.1007/978-3-030-79725-6_64
- Di Martino B, Marulli F, Lupi P, Cataldi A (2021c) A machine learning based methodology for automatic annotation and anonymisation of privacy-related items in textual documents for justice domain. In: Barolli L, Poniszewska-Maranda A, Enokido T (eds) *Complex, intelligent and software intensive systems*. Springer International Publishing, Cham, pp 530–539. https://doi.org/10.1007/978-3-030-50454-0_55
- Di Martino B, Colucci Cante L, D'Angelo S, Esposito A, Graziano M, Marulli F, Lupi P, Cataldi A (2021d) A big data pipeline and machine learning for a uniform semantic representation of structured data and documents from information systems of Italian ministry of justice. *Int J Grid High Perform Comput (IJGHPC)* (in press)
- Di Martino B, Colucci Cante L, Esposito A, Lupi P, Orlando M (2021e) Temporal outlier analysis of online civil trial cases based on graph and process mining techniques. *Int J Big Data Intell (IJBDI)*. <https://doi.org/10.1504/IJBDI.2021.10037170> (in press)
- He BY, Zhou J, Ma Z, Wang D, Sha D, Lee M, Chow JY, Ozbay K (2021) A validated multi-agent simulation test bed to evaluate congestion pricing policies on population segments by time-of-day in new york city. *Transp Policy* 101:145–161
- Logenthiran T, Srinivasan D, Khambadkone A, Aung H (2010) Multi-agent system (mas) for short-term generation scheduling of a microgrid. In: *2010 IEEE international conference on sustainable energy technologies (icset)*. IEEE, pp 1–6
- Lupi P (2017) Il processo civile realizzato con strumenti telematici. Technical Report, Superior School of Magistracy
- Massimo Orlando GV (2020) Il controllo di gestione negli uffici giudiziari: il “laboratorio” livorno. *Questione Giustizia Trimestrale promosso da Magistratura democratica—Fascicolo 1 Eguaglianza e diritto civile*
- Mishra SK, Singh VK (2017) Developing a multi agent system model in gaia for court case management system as a case study and providing an extension to gaia. In: *2017 3rd international conference on advances in computing, communication automation (ICACCA) (Fall)*, pp 1–4. <https://doi.org/10.1109/ICACCA.2017.8344742>
- Simone V, Picilli G, Piazzolla G (2013) Its: un approccio multi agente basato su jade. Masters thesis, Università degli Studi di Bari Aldo Moro
- Stranjak A, Dutta PS, Ebdem M, Rogers A, Vytelingum P (2008) A multi-agent simulation system for prediction and scheduling of aero engine overhaul. In: *Proceedings of the 7th international joint conference on autonomous agents and multiagent systems: industrial track, international foundation for autonomous agents and multiagent systems, Richland, SC, AAMAS 08*, pp 81–88
- Wooldridge M, Jennings NR, Kinny D (2000) The gaia methodology for agent-oriented analysis and design. *Auton Agent Multi Agent Syst* 3(3):285–312
- Yue T, Long R, Chen H, Liu J, Liu H, Gu Y (2020) Energy-saving behavior of urban residents in china: a multi-agent simulation. *J Clean Prod* 252:119623
- Zhong Q, Luo X, Eemeren FH, Huang F (2013) The task model of court investigation in a multi-agent system of argumentation in court. In: *Proceedings of the 4th international workshop on logic, rationality, and interaction*, vol 8196, LORI 2013. Springer, Berlin, pp 296–310. https://doi.org/10.1007/978-3-642-40948-6_23
- Zhu J, Shi J, Yang Z, Li B (2019) A real-time decentralized algorithm for task scheduling in multi-agent system with continuous damage. *Appl Soft Comput* 83:105628. <https://doi.org/10.1016/j.asoc.2019.105628>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.