Decision Support Based on Human-Machine Collaboration Patterns: Conceptual Model and Scenario

Alexander Smirnov, Tatiana Levashova
St. Petersburg Federal Research Center of the Russian Academy of Sciences
St. Petersburg, Russia
smir, tatiana.levashova@iias.spb.su

***Abstract*—The paper proposes a conceptual model of human-machine collaborative decision support in which humans and software agents use collaboration patterns to achieve goals of decision-making process steps. An analysis of collaboration patterns identified in multiple domains is provided and a pattern classification is offered. A sample scenario for collaborative decision support is proposed. The scenario illustrates a possible functionality of a collaborative decision support system implemented according to the conceptual model developed.**

I. INTRODUCTION

Nowadays, role of machines (software intelligent agents, artificial intelligence) as a collaborative partner of humans has become pivotal. When humans work in collaboration with machines, they harness the power of the machine's capabilities to enhance their decision-making and problem-solving processes. Pattern is a description of a reusable efficient and proven solution to a problem that occurs repeatedly in a variety of contexts. Collaboration pattern proposes a solution for collaborative problem solving / goal achievement by providing proven ways of organizing communication and joint activities for specific collaboration tasks.

The paper presents an innovative approach that combines human expertise with software agents for decision support purposes and facilitate collaboration of humans and agents by providing them with patterns that offer reusable solutions for collaborative problems occurring at different decision-making steps.

Mostly, collaboration patterns inherit the idea of the design patterns proposed by Christopher Alexander [1] in the 70s to systematize solutions for problems recurring in architecture (253 patterns). Each pattern describes a problem, which occurs over and over again, and proposes a reusable solution for this problem. In architecture, the design patterns have not found a widespread usage, but the idea of these patterns has attracted the attention of other domains including design, software development, and collaboration.

Although in this paper, collaboration comprises a group of humans and machines, the paper analyses collaboration patterns identified in multiple domains not only in relation to human-machine collaboration, but also to the collaboration of humans or software agents (machines). The analysis forms the base for the development of a conceptual model of decision support based on human-machine collaboration patterns (further, the Conceptual model).

As a result of the pattern analysis, the collaboration patterns have been classified into 4 categories: process patterns, collaborative engineering patterns, cognitive patterns, and interaction patterns. For each category, distinctive features of the patterns that belong to a category and common elements of pattern representations are identified. The distinctive features provide ideas on conditions when the patterns of a category may be used. The common elements allow ones to define which concepts to introduce in the Conceptual model. The paper does not provide complete lists of concepts used to represent the elements of the patterns analyzed, but offers ready-made results of their analysis. The lists are a technical aspect, the inclusion of them results in paper clutter.

The rest of the paper is organized as follows. Collaboration patterns are analyzed in Section II in accordance with the proposed classification. Section III describes the Conceptual model and proposes a sample scenario for this model. Section IV summarizes the main results, provides some concluding remarks and discusses possible future research.

II. COLLABORATION PATTERNS

A. Process patterns

Process patterns describe a problem solution in terms of activities, actions, and work tasks that the collaborators must take or accomplish to come to this solution as well as tools they can use.

Collaboration patterns from the domain of virtual organizations are a kind of the process patterns [2], [3]. Collaboration here is defined as joint actions of the collaborators to solve collaborative problems that may occur repeatedly in the environment. The collaborators are supposed to be humans and machines. A collaboration pattern prescribes the manners of collaboration and describes problem solutions in the form of a recurring group of actions, workflows and instructions for tool usage. For the pattern representation, a pattern model in the form of a set of elements is used. These elements are common for all the patterns built based on the pattern model. When the elements are instantiated, a pattern instance is produced. Solution is represented by diagrams in Business Process Management Notation (BPMN) [4]. The

number of collaborative problems determines the number of patterns. Use cases for the collaboration pattern in virtual organizations cover collaboration in various domains including manufacturing, education, or construction. For example, based on the model a set of collaboration patterns is developed, which comprises collaboration patterns for completing discussion of a draft report, inter-organizational knowledge integration, tactical and strategic recommendations to improve the business of the virtual organization, and completing a client's order in supply chain environment [5].

Collaboration patterns for software process development [6] belong to the category of process patterns. Here, collaboration is defined as a process that contains at least one collaborative activity being performed by two or more human actors targeting the same goal. Collaborative activity is a coordinated and synchronous task whose goal is to build and maintain a shared design of a problem. The collaboration patterns [6] are based on workflow patterns [7], which represent reusable generic fragments of the process. The workflow patterns is a set of 42 generic patterns grouped into 8 parts. From this set, the authors [6] chose a number of patterns that meet the idea of collaboration. Collaboration pattern is referred to as a recurrent problem, a solution and an application context. A pattern is represented as concepts and relationships defined in the CMSPEM metamodel. CMSPEM is a process modelling language, which is an extension of the OMG standard SPEM (System and Software Process Engineering Metamodel) [8] for describing collaborative software processes. Solution that the patterns propose is a workflow represented in the form of an activity diagram in CMSPEM. Uses cases of collaboration patterns for software process development include the patterns "Duplicate in Sequence with Multiple Actors" and "Duplicate in Parallel with Multiple Actors and Merge" used to solve the problem "Review a Deliverable" performed during a project.

Team design patterns [9] complete the review of process patterns. These patterns are characterized as a design tool for human-machine teaming. Six elementary types of work underlie teaming patterns. Physical and cognitive types of work are distinguished. Each of this type can be of direct, indirect and off-task work. Possible combinations of work in which teammates can be involved are i) physical direct, indirect, and off-task work, and ii) cognitive direct, indirect, and off-task work. Joint work is a set of direct and/or indirect work carried out by two or more teammates related to the same common goal. An ontology is developed to represent team patterns for various types of joint work. Solution proposed by the team design patterns is a large team pattern for the collaborative task. Such a pattern is a network of small patterns (or sub-patterns), where the network's links represent transitions between the sub-patterns. The representation with a network is the reason why the team design patterns are classified as the process patterns. Use case of team design patterns describes supervisory control team patterns.

Summary on the process patterns

- *Collaboration process* consists in taking actions by collaborators leading to problem solution or goal achievement.
- *Objective* is building collaboration process.
- *Solution* is a sequence of actions or patterns for actions, which shapes collaborative scenario.
- *Pattern representation*: a set of elements [2], [3], meta-model of process modelling language [6], ontology [9].
- *Solution representation*: process representation diagrams.
- *Common elements* of the pattern models are intended to specify a recurrent problem/goal and activities to solve the problem / achieve the goal. Table I provides names for the concepts used to represent the pattern elements.
- *Common elements* of the pattern models in terms of the Conceptual model are *Problem* and *Activity* (Table II).

TABLE I. PROCESS PATTERN MODELS: COMMON CONCEPTS

Element purpose	Source	Concept name	Definition
Specification of a goal/ problem	[2], [3]	Problem	A description of the problem(s) the collaboration pattern has addressed before or it is expected to address in the future
	[6]	Task	A reusable task aiming at the use, development, or modification of the work product
	[9]	Example	A description of a collaborative scenario in terms of kinds of teammates and activities that they perform
Specification of activities to achieve the goal/ solve the problem	[2], [3]	Category	Objectives and functions of a virtual organization supported by the pattern
	[6]	Activity	Work performed by a role responsible for doing it and resulting in a work product
	[9]	Work	Activity that an individual or team undertakes in order to achieve the team goal

TABLE II. NAMES AND DEFINITIONS OF PROCESS PATTERN CONCEPTS IN THE CONCEPTUAL MODEL

Concept name in source	Concept name in the Conceptual model	Definition in the Conceptual model
Problem, Task, Example	Problem	A reusable problem/goal that participants jointly solve/achieve
Category, Activity, Work	Activity	Behavior of participants in the process of collaborative problem solving /goal achieving

B. Collaborative engineering patterns

The collaborative engineering patterns describe a problem solution in terms of rules supporting the creation of collaboration or rules for decision making by interested stakeholders. This collection is represented by two kinds of patterns: collaborative decision making [10], [11] and organization of collaboration [12], [13].

The collaborative decision-making defines collaboration patterns as techniques, behaviors, and activities for people who share a common goal of working together in a group [10]. The first pattern in the group of the collaborative decision-making patterns is the pattern "decide how to decide" [10]. This pattern addresses the problem that participants of a decision-making

process often do not know a rule according to which they will come to a common decision. The pattern proposes a problem solution by establishing a procedure for choosing a decision-making rule for a given context. Context is a specific issue that should be closed, that is on which a decision should be made. Eight common decision-making rules are provided for. They are majority vote, delegation, negotiation, spontaneous agreement, arbitrary, decision leader decides without discussion, decision leader decides after discussion, and consensus. The pattern is represented by a set of elements. The solution is represented by an ordered list of textual rules how to choose a decision rule from the eight common ones.

The other pattern from the group of the collaborative decision-making patterns is referred to as "collaborative decision making" [11]. It is proposed for collaboration activities in Enterprise 2.0. Enterprise 2.0 is a system of Web-technologies enabling the users (employees, partners, suppliers, and customers) to work together, share information, and make group decisions through the use of Web 2.0 communication tools within a company. The pattern allows including several actors in a collaboration activity where the goal is to take a decision about some topic and the decision involves several responsibilities. A solution proposed in the pattern is a workflow representing roles and activities undertaken by humans fulfilling these roles to come to a decision. The pattern is represented as a set of elements where the element "Structure" that defines the business process flow is represented by means of BPMN diagrams [4]. Solution is represented by means of these diagrams, also.

With relation to collaboration organization, collaboration patterns are defined as a set of objects, actions, rules, and steps for participants with roles who meet at a location to collaborate on a common goal in a given context [12]. Organization of collaboration is an objective of the framework of collaboration in virtual environments [12]. The framework serves as a blueprint to guide users in designing, implementing, and executing virtual collaboration patterns (such as planning, evaluation, decision making or debriefing) tailored to their needs. Collaboration takes place in a virtual environment that is defined as "a synchronous, persistent network of people, represented as avatars, facilitated by networked computers". The patterns pursue the idea of best benefit from meeting with colleagues and peers in a virtual environment with the aim of working together. A pattern is represented by a set of elements. The elements of the pattern model are intended to indicate a context (collaborate, learn, or play); declare the collaboration goal; specify where, when, whom and how a certain pattern should be used; and to describe the infrastructure of the virtual environment. The goals and contexts are related (e.g., for the context "collaborate", goals of share, design, evaluate, etc. can be defined). The infrastructure is made up of sets of actions and objects. A proposed solution is a specific collaboration pattern that is an instance of the framework (the pattern model) defined using the parameters positioned within the framework.

Human Systems Dynamics Institute (HSD), USA (www.hsdinstitute.org), which was found in 2003 to develop and propagate the theory and practice of self-organizing multi-scale human systems based on complexity theory, proposes four patterns for organization of collaboration [13]: Mutual Support, Shared Projects, Joint Ventures, and Strategic Partners. Mutual Support is collaboration of individuals aiming at knowledge exchange between colleagues of different areas of expertise. Shared Projects is collaboration of organizations indented to exchange information that can be used in projects of these organizations. Joint Ventures is collaboration engaging colleagues into groups to pursue together a large-scale initiative or program. Strategic Partners is collaboration of organizations that brings the partners closer together than Shared Projects and Joint Ventures and that characterized by shared strategic partner identities. The addressed patterns determine the goals of the collaboration (clear from the patterns names), the structures of partner interactions, and kinds of activities related to a specific purpose. The partners make a decision on a particular pattern when they are planning their collaboration. The "right" pattern depends on the purpose, context, history, and players. The chosen pattern is actual as long as the partners agree to the collaboration purpose. The patterns are represented by a set of elements. A "solution" element is not provided for and rules how to choose a collaboration pattern are not specified explicitly.

Summary on the collaborative engineering patterns

- *Collaboration process* is a decision-making process or activities in accordance with the principles and structure of the collaborative environment.
- *Objective* is proposing a collaborative environment or collaborative decision-making rules.
- *Solution* is a pattern instance representing rules [10] or procedure [11] how to make a collaborative decision, a framework of collaborative environment [12], or characteristics of the collaboration [13].
- Pattern representation: a set of elements.
- *Solution representation*: a pattern instance, workflow.
- *Common elements* of the pattern models are intended to specify the current state of anything of collaboration interest (e.g., entities, events, ongoing processes and characteristics of these entities, events, and processes); a recurrent problem /goal; and activities, actions, and interactions with relation to the current state (Table III).
- *Common elements* of the pattern models in terms of the Conceptual model are *Context*, *Problem*, and *Activity* (Table IV).

In Table III and Table VII, the wording "referred to as Concept" in the column "Concept name" means that the name of the concept appears in the source description of the pattern explicitly, but is not included in the pattern structure/model (the concept may or may not have a definition); the wording "close to Concept" means that the semantic meaning corresponding to the "Concept" is found in the source description of the pattern, but has no name, is not explicitly defined and is not included in the structure/model of the pattern. In this regard, the definitions (column "Definition") for the concepts not explicitly defined in the sources are formulated by the authors of this paper based on the analysis of the source text.

TABLE III. COLLABORATIVE ENGINEERING PATTERNS: COMMON CONCEPTS

Element purpose	Source	Concept name	Definition
Specification of the current state of anything of collaboration interest	[10]	Context	Making a group aware about closure on a specific issue, when and whether a decision is made
	[11]	Not defined, referred to as Problem	Identifying and designing collaboration processes in order to integrate them in the information system
	[12]	Goal	A kind of activity defined for a specific context
	[13]	Goal	A characteristic of partner goals (different when each partner pursues own goal, project-level, initiative-level, organizational missions)
Specification of a goal/ problem	[10]	Problem	Explicit formulation of the "rule" for how a group will reach decisions
	[11]	Not defined, referred to as Goal	Take a decision about some topic and the decision involves several responsibilities
	[12]	Not defined, close to Problem	Formalizing collaboration in virtual environments
	[13]	Not defined, close to Problem	Making a decision about a collaborative pattern
Specification of activities, actions, interactions	[10]	Uses	Kinds of activities for which a pattern can be used
	[11]	Not defined, referred to as Activity	A kind of collaboration activity in a team that must work in a collaborative way
	[12]	Actions	Kinds of object interactions (communication, navigation, and manipulation)
	[13]	Activities	Kinds of activities that a pattern supports

TABLE IV. NAMES AND DEFINITIONS OF COLLABORATIVE ENGINEERING PATTERNS CONCEPTS IN THE CONCEPTUAL MODEL

Concept name in source	Concept name in the Conceptual model	Definition in the Conceptual model
Context, Goal, Problem	Context	Situation in which a pattern can be used
Problem, Goal	Problem	A reusable problem/goal that participants jointly solve/achieve
Uses, Activity, Activities, Actions	Activity	Behaviour of participants in the process of collaborative problem solving / goal achieving

C. Cognitive patterns

Cognitive patterns describe the thinking and reasoning processes of experts. This group is represented by two patterns: thinking patterns [14], [15] and patterns of inference for intelligence analysis [16].

The model thinkLet [14], [15] supports general descriptive patterns (thinkLets) of thinking in performing an intellectual task collaboratively. Each thinkLet addresses a particular pattern of collaboration, that is, a generic activity that teams need to undertake in order to accomplish collaborative tasks. The set of thinkLets is made up of six patterns: generate, reduce, clarify, organize, evaluate, build consensus. Generate means to move from having fewer to having more concepts in the pool of concepts shared by the group; an example of using is brainstorming ideas on a focused set of topics simultaneously. Reduce is movement from having many concepts to a focus on fewer concepts that the group deems worthy of further attention; an example of using is extracting of a list of key ideas from a raw set of brainstorming comments. Clarify is about movement from having less to having more shared understanding of concepts and of the words and phrases used to express them; an example of using is making sure that the team members agree on the meaning and phrasing of the items on the resultant list. Organize supposes movement from less to more understanding of the relationships among concepts the group is considering; an example of using is organizing a large set of ideas into categories. Evaluate aims at movement from less to more understanding of the relative value of the concepts under consideration; an example of using is evaluating a number of ideas with respect to one or more criteria. Build consensus means to move from having fewer to having more group members who are willing to commit to a proposal; an example of using is continuously tracking the level of consensus within the group with regard to a particular issue. Solution implied by the thinkLet model suggests typical combinations of a thinkLet with other thinkLets for collaboration engineers when developing a script (a bare-bones example of the instructions a practitioner or facilitator could give to the group to create the desired group interactions). A process diagram is used to represent a solution. The diagram provides a sequence of thinkLets for a specific collaboration scenario.

The aim of intelligence analysis is to make sense of information (often conflicting or incomplete) to explain an observed situation [16]. Weighing up competing hypotheses is the techniques that human analysts use to this aim. Human-machine collaboration in this area is that experts put forward hypotheses and artificial intelligence (AI) provides automated support to identify what claims and pieces of evidence can together form a plausible hypothesis, and what other alternatives exists that are also plausible, by employing computational models of argumentation. The authors [16] consider inference patterns as collaboration ones. The inference patterns is an argument scheme. They represent templates for making presumptive inferences formed by premises supporting a conclusion, and by critical questions that can be put forward against the applicability of the inference. A pattern is represented as a propositional graph implementing the following structure: (a set of) assumption(s) → conclusion (hypothesis). Solution that the patterns propose is sets of elements for argumentation (entity, agent, fact, activity), relations among these elements (causal or associative), and critical questions relevant to the argumentation scheme. The solution has the form of an argumentation scheme representing human-readable propositional calculus formulas. The patterns are executable. Solution provided in result of automatic pattern execution is a set of plausible hypotheses. Two patterns are provided as examples: 1) *argument scheme from cause to effect* to provide an explanation for some set of observations on the basis of activities and events that shows how the situation has evolved; and 2) *argument for identifying an agent from past*

actions which encodes the sensemaking process that shifts from understanding what happened to understanding what entities were involved and their association with the activity.

Summary on the cognitive patterns

- *Collaboration process* is putting forward ideas, proposals, hypotheses and their agreeing.
- *Objective* is to shape a scenario for putting forward ideas, proposals, hypotheses and their agreeing.
- *Solution* is a configuration of patterns or pattern elements to organize the scenario of the collaboration process.
- *Pattern representation*: pattern model [14], [15], logical propositions [16].
- *Solution representation*: workflow [14], [15], human-readable formulas of propositional calculus extended with critical questions [16].
- *Common elements* of the pattern models are intended to specify responsibilities of collaborators according to their roles; collaborators' activities; and rules according to which the activities are undertaken or evaluated (Table V).
- *Common elements* of the pattern models in terms of the Conceptual model are *Role*, *Activity*, and *Rule* (Table VI).

TABLE V. COGNITIVE PATTERNS: COMMON CONCEPTS

Element purpose	Source	Concept name	Definition
Specification of responsibilities of actors/participants according to their roles	[14], [15]	Role	A position of a participant or a group according to which they are responsible for carrying out individual or group tasks (activities), respectively
	[16]	Agent, Source	• Agent is something or someone responsible for an activity taking place such as a person, or a software tool • Source is a domain expert containing a proposition
Specification of activities that actors/participants carry out	[14], [15]	Activities	Individual or collaborative tasks sequencing of which results in a collaborative process
	[16]	Activities	Actions performed by actors and events happening in the world
Specification of rules for action execution and evaluation of argumentation conclusions	[14], [15]	Rules	Actions that participants must execute using the capabilities provided to them under some set of constraints
	[16]	Rule	A link between an activity and its effect that also can be an activity

D. Interaction patterns

Interaction patterns describe collaboration supported by information systems (applications, social networks, on-line communication systems, etc.) They specify a problem solution in terms of components of communication and data management processes.

TABLE VI. NAMES AND DEFINITIONS FOR CONCEPTS OF COLLABORATIVE ENGINEERING PATTERNS IN THE CONCEPTUAL MODEL

Concept name in source	Concept name in the Conceptual model	Definition in the Conceptual model
Role, Agent, Source	Role	A position of participants according to which they are responsible for carrying out activities
Activities	Activity	Behaviour of participants in the process of collaborative problem solving / goal achieving
Rules, Rule	Rule	Actions that participants have to undertake using the capabilities provided to them under some set of constraints

Interaction patterns are part of the set of patterns [11] proposed to manage information and knowledge that users of Enterprise 2.0 exchange. Here, the interaction patterns are represented by the patterns "Collaborate", "Collaborate Enhanced", and "Aggregate Activity Loop". These patterns are intended to trace and elicit knowledge from the information exchanged. The patterns "Collaborate" and "Collaborate Enhanced" aggregate information exchanged while collaborating. Compared to "Collaborate", the "Collaborate Enhanced" pattern has the option of storing the collaboration activity or of discarding it. The goal of "Aggregate Activity Loop" is to store information coming from collaboration activities that use web 2.0 tools. The patterns are represented as a set of elements where the element "Structure" that defines the business process flow is represented by means of BPMN diagrams [4]. Solution is represented by means of such diagrams, also.

Collaboration patterns in (human) communities [17] can be thought of as a sort of interaction patterns. The model of these patterns is defined as reusable conceptual structures capturing essential collaboration requirements. The collection of the patterns includes goal patterns, communication patterns, information patterns, task patterns, and meta-patterns. The goal patterns describe what the collaboration is about; these patterns are conceptual representations of community & individual objectives. The communication patterns describe how communication to accomplish the goals takes place; the patterns are sets of related communicative workflow and norm definitions describing acceptable and desired communicative interactions within a community. The information patterns define the knowledge elements essential for the collaborative process; the patterns are conceptualizations of content knowledge obtained from knowledge analysis activities. The task patterns define which information patterns are to be created in particular steps in the communicative process, thus describing the role that content can play in collaborative communication. The meta-patterns are conceptual patterns necessary to interpret, validate, link, and assess the quality of other collaboration patterns. A representation by the formalism of conceptual graphs [18] is developed for each type of the patterns from the collection. Solution that the patterns propose is a conceptual graph representing a composite collaboration pattern for a specific scenario. The patterns for collaboration in communities were applied for designing a socio-technical infrastructure of collaborative communities for social

innovations [19] and organizing collaboration of professionals through an online system conducting meetings within a medicine society [20]. In this society, the original representation of the patterns and solutions with the conceptual graphs is replaced with an OWL-based representation [21]. This is partly because the online system for meetings is based on the Semantic Web technologies OWL and RDF.

One more example of the interaction patterns are patterns of collaboration in socio-technical systems [22]. In these systems, collaboration patterns are considered as "human architectures" consisting of people (human components) and the systems they use to facilitate collaboration (collaboration connectors). The pattern model provides reusable structures of humans as components of the socio-technical systems and their roles, collaboration connectors, actions (interactions amongst components and connectors in terms of Create, Read, Update, and Delete manipulation capabilities), collaboration objects (messages (Task, Job, Result), streams, and artifacts (Outcome)), and action relationships. Connecting human actions and matching object actions gives rise to a collaboration pattern. In order to create simulation models for human collaboration, the authors extend the human Architecture Description Language (hADL) [23]. The extended hADL can be used to develop a model of a collaboration pattern or composition of patterns, define the structure and behaviour of the pattern(s), and create simulation models for human collaboration. This language is used for patterns representation. Solution proposed by the patterns it is a composite pattern in the extended hADL language for a specific scenario; it describes what kind of human components and coordinators have proven suitable for a given joint effort.

Summary on the interaction patterns

- *Collaboration process* is exchanging and editing information, and performing procedures and tasks initiated by information messages.
- *Objective* is to provide elementary patterns to organize the scenario of the collaboration process.
- *Solution* is a composite collaboration pattern for a specific scenario.
- *Pattern representation*: workflow modelling diagrams [11], [23], ontology [17].
- *Solution representation*: workflow formalized by pattern representation means.
- *Common elements* of the pattern models are intended to specify interactions among the collaborators and tools that support such interactions; responsibilities, duties, and authorities of collaborators according to their roles; collaborators themselves; and collaborators' activities.
- *Common elements* of the pattern models in terms of the Conceptual model are *Tool*, *Participant*, *Role*, *Activity* (Table VII, Table VIII).

III. Decision support based on human-machine collaboration patterns

The Section describes the Conceptual model developed, provides ideas on the origin of concepts introduced in this model, discusses the model semantics, and proposes a sample scenario to illustrate a model functionality.

TABLE VII. Interaction patterns: common concepts

Element purpose	Source	Concept name	Definition
Specification of tools supporting interactions among actors/ community members and manners of messaging	[11]	Not defined, referred to as Tool	A web 2.0 communication tool
	[17]	Not defined, referred to as Tool and Instrument	A tool augmenting people in retrieving relevant community knowledge
	[22]	Collaboration Object	Collaboration elements as messages, streams, or shared artifacts without specifying a communication means
Specification of responsibilities, duties, and authorities of actors/ community members according to their roles	[11]	Not defined, referred to as Role	A position of an actor responsible for carrying out tasks supposed by this position
	[17]	Not defined, referred to as Role	A position of a community member regulating the member behaviour by applying community-defined norms and rules that prescribe what regulation behaviour may, must, or may not be performed by the member in this position
	[22]	Role	A name for a set of activities for fulfilling those human, hardware, or software is responsible (work-centric roles and coordination-centric roles are distinguished)
Specification of collaborators	[11]	Not defined, referred to as Actor	One or more team members that fulfill a role involved in a collaboration scenario
	[17]	Not defined, referred to as Community member	A human resource of a collaborative community
	[22]	Not defined, referred to as Team member	Users and user groups that fulfill a role involved in collaboration among members of a team
Specification of individual and collaboration activities, and kinds of interactions	[11]	Not defined, referred to as Activity	A kind of collaboration activity in a team that must work in a collaborative way
	[17]	Not defined, close to Activity	An individual activity supposed by a specific collaboration pattern
	[22]	Actions	Interactions amongst humans and systems that facilitate collaboration

A. Conceptual model of decision support based on human-machine collaboration patterns

The development of the Conceptual model starts with the identification of concepts common for different pattern representations. For this, concepts used to name elements in existing collaboration patterns models are analyzed and generalized (one name is chosen for semantically similar concepts named differently in different representations). The analysis has shown that *Activity* is the concept common for all the reviewed collaboration patterns. It is defined as behavior of collaborators in the process of collaborative problem solving / goal achieving.

TABLE VIII. NAMES AND DEFINITIONS OF INTERACTION PATTERNS CONCEPTS IN THE CONCEPTUAL MODEL

Concept name in source	Concept name in the Conceptual model	Definition in the Conceptual model
Tool, Instrument, System, Collaboration Object	Communication tool	A tool that supports communications between participants
Actor, Community member, Team member	Participant	Human or software agent involved in collaboration
Role	Role	A position of a participant responsible for undertaking activities supposed by this position
Activity, Actions	Activity	Behaviour of participants in the process of collaborative problem solving / goal achieving

The following concepts are introduced in the Conceptual model (Fig. 1) for its consistency.

Participant is a human expert or an intelligent software agent engaged in collaboration and carrying out activities supposed by the roles that they fulfill. This concept is introduced because the concept of *Activity* is defined through the concept of *Participant* (in various representations, teammate, partner, collaborator, actor, or member).

Role is a position of participants according to which they are responsible for carrying out activities. This concept is introduced because the definition for the concept of *Participant* uses the concept of *Role*.

Problem is a task/objective that participants solve/achieve collaboratively. This concept is introduced because the concept of *Activity* is defined through the concept of *Problem*. Problem can be a task/objective that the user of the decision support system (DSS) deals with or a pattern goal.

Task is a task/objective that the user formulates in his/her request to the DSS and for which a collaborative team is organized. A task is characterized by its current processing status. The concept is introduced according to the definition of the concept of *Problem*.

Goal is a recurring collaborative problem for which a collaboration pattern proposes a reusable solution. The concept is introduced according to the definition of the concept of *Problem*.

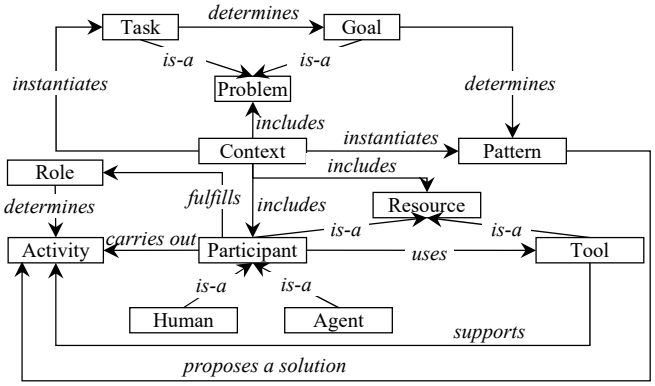

Fig. 1. Conceptual model of decision support based on human-machine collaboration patterns

Concepts below are introduced to ensure context-awareness of the DSS.

Context is a situation in which a specific collaboration pattern can be used. Context is characterized by *Task*, *Goal*, and *Resources*.

Resource is an available source of aid or support that may be drawn upon when needed. *Participant* and *Tool* are kinds of resources.

Tool is a means that supports an activity (e.g., a word processor supports a document editing activity).

The Conceptual model supposes a decision support scenario where the decision maker formulates a task to the DSS, which is a system wherein multiple software agents and humans collaborate. The system processes this task as a decision-making problem [24]. A result of problem solving is a recommended decision or a set of alternatives (generally, may be empty) accompanied with supporting information for making an informed decision.

The semantics behind the Conceptual model (Fig. 1) is as follows. In some context, the decision maker via the request to the DSS formulates a task that he/she is dealing with. The context contains information about the available resources and the collaborative problem. The resources are collaboration participants (humans and agents) and tools. The collaborative problem represents tasks and collaboration goals. The context instantiates the tasks by creating a task instance that represents the task formulated by the decision maker. The participants process the task collaboratively as a decision-making problem using appropriate patterns at different processing stages. The processing status of the task determines a current collaboration goal. This goal determines a collaboration pattern that the participants can use in the task processing process. The collaboration pattern proposes activities that the participants are expected to carry out in accordance with their roles. They can use tools that support these activities. The context produces instances necessary for pattern executions.

The conceptual model supposes incorporation of up-to-date artificial intelligence techniques such as speech recognition, natural language processing, predictive analytics, and explainable artificial intelligence. Speech recognition and natural language processing support user request recognition, task identification, and creation of a machine-readable task specification. The focus of predictive analytics is to provide personalized decision support and choosing collaboration patterns taking into account personal human traits and individual behavior models. Explainable artificial intelligence is a power tool to support human-machine collaboration by providing humans with explanations of the agents' decisions.

B. Scenario of decision support based on human-machine collaboration patterns

The scenario of collaborative decision support (Fig. 2) describes the following activities.

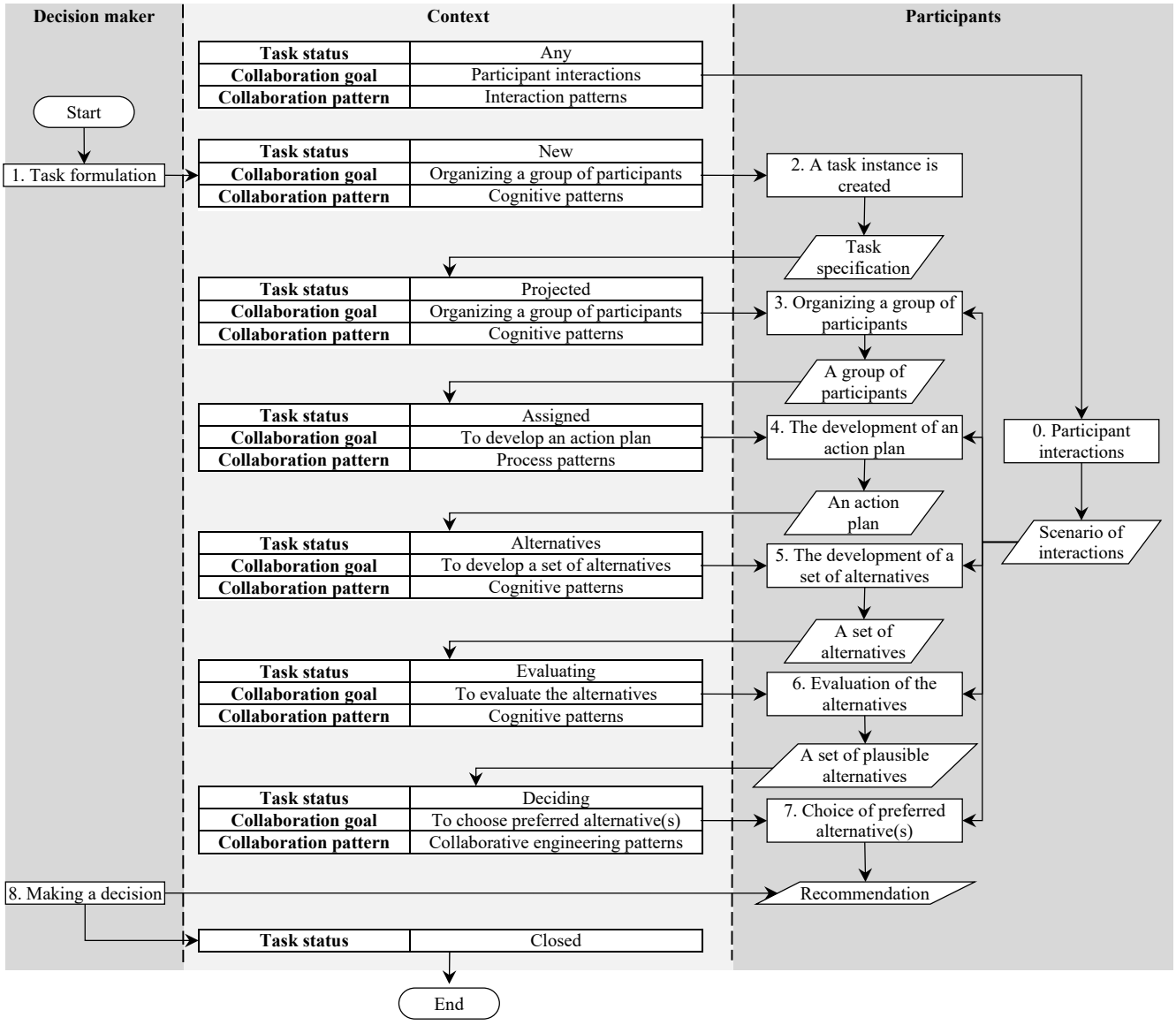

Fig. 2. A sample scenario for collaborative decision support

1. The decision maker formulates a task. The task gets the status of "New".
2. The context creates a task instance and assigns the task status of "Projected". This status means that the current collaboration goal is organizing a group of participants for task processing.
3. Humans and agents available in the context participate in the organizing a human-agent collaborative group. As one of the ways to achieve the collaboration goal they can use cognitive patterns proposing a solution in the form of a scenario for putting forward ideas and proposals and agree them. As a result of the scenario execution, a group of participants is organized, which comprises humans and agents who have agreed with each other on their collaboration. The context changes the task status to "Assigned", which means that the participants start to process the task and the collaboration goal is to develop an action plan.
4. The participants develop an action plan. For this purpose, they use process patterns. These patterns provide a solution in form of a set of activities, roles responsible for their undertaking, and tools needed to complete them. As soon as the action plan is developed, the context changes the task status to "Alternatives". This status prescribes the collaboration goal as the development of alternatives.
5. In the process of developing alternatives, the participants use cognitive patterns, which provide a scenario for putting forward hypothesis. When a set of alternatives has been developed, the context changes the task status to "Evaluating". The collaboration goal corresponding to this status is evaluation of the alternatives.
6. A pattern offering a solution for the goal of alternatives evaluation is a cognitive pattern that proposes an argumentation scheme to form plausible hypotheses (here, alternatives). After the goal of evaluation is achieved, the context changes the task status to "Deciding". This status

prescribes the collaboration goal as choice of preferred alternative(s).

7. Collaborative engineering patterns offer a solution to achieve the goal relating to the choice of preferred alternative(s) by providing collaborative decision-making procedure /rules.
8. The proposed scenario implies that the chosen alternative is a recommended decision, which is delivered to the decision maker and he/she makes a decision on either to accept or to decline it. After the decision maker has received the recommendation, the context changes the task status to "Closed", which means that the scenario is completed.
0. Any task status supposes existence of the goal which focuses on participant interactions. Interactions patterns are a means to achieve this goal. They offer a solution in the form of a scheme for participant interactions in a specific scenario.

The scenario proposed is not the only possible. For instance, the goal of organizing a group of participants can be achieved using team design patterns as a kind of process patterns, or collaborative engineering patterns that propose a procedure for collaborative decision-making, which can be applied to organize a collaborative environment based on the characteristics of such an environment and characteristics of collaboration. Moreover, just as the interactions patterns, collaborative engineering patterns providing for collaborative decision-making procedure /rules can be needed anywhere a decision is required. Other scenarios are also possible. Specification of details of which pattern can be used under given circumstances is a possible direction of the future research.

VII. Conclusion

Collaboration patterns offer proven reusable solutions for collaborative activities thereby increasing the efficiency of collaboration. The paper exploits the idea of collaboration patterns to propose a conceptual model for human-machine collaborative decision support. In this model, the participants of collaboration (software agents and humans) use collaboration patterns to achieve goals specified in context. These goals correspond to step objectives of decision-making process.

The conceptual model is built based on the results of the analysis of collaboration patterns identified in multiple domains provides. An expectation from this analysis was a set of concepts, which are common for naming pattern representation elements in existing collaboration patterns models and in meta-models of pattern modelling languages. Actually, the analysis has shown that only the concept of Activity is common for all the analyzed patterns. This concept is introduced in the conceptual model along with other concepts allowing for a consistent model and context-aware decision support.

A sample scenario for collaborative decision support is proposed. The scenario illustrates a possible functionality of a DSS implemented according to the conceptual model developed.

The main drawback concerns situations when different patterns can be used to achieve the same goal. So far, it is a lack of details, which pattern is better to use to achieve a specific collaboration goal. This drawback guides the future research. Some other directions of the future research are evaluation of the quality of collaboration and prototyping the conceptual model proposed using real-world scenarios. The collaboration quality is planned to evaluate by metrics that can be used to assess how well the current goals are defined, how well humans understand the intentions and decisions of the agents, how humans are satisfied with the efforts that they have made, how a pattern proposed goes well with the human traits and individual behavior, and how the user is satisfied with the recommendation. For the prototyping, the human-machine collective intelligence environment for decision support [25] is planned to be used, which is currently under development.

Acknowledgment

The paper is due to the State Research Project no. FFZF-2022-0005.

References

[1] C. Alexander, S. Ishikawa, and M. Silverstein, *A pattern language*. Oxford: Oxford University Press, 1977.
[2] N. Papageorgiou, Y. Verginadis, D. Apostolou, and G. Mentzas, "A collaboration pattern model for virtual organisations," in *Leveraging Knowledge for Innovation in Collaborative Networks. PRO-VE 2009*, vol. 307, L. M. Camarinha-Matos, I. Paraskakis, and H. Afsarmanesh, Eds. Berlin Heidelberg: Springer, 2009, pp. 61–68.
[3] Y. Verginadis, D. Apostolou, N. Papageorgiou, and G. Mentzas, "An architecture for collaboration patterns in agile event-driven environments," in *2009 18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, Jun. 2009, pp. 227–230, doi: 10.1109/WETICE.2009.12.
[4] P. Briol, *BPMN, the Business Process Modeling Notation*, 1st ed. Lulu.com, 2008.
[5] C. Palmer *et al.*, "A demonstration of collaboration pattern applications in virtual organizations," in *The 7th International Conference on Manufacturing Research (ICMR09)*, 2009, pp. 193–197.
[6] T. T. Vo, B. Coulette, H. N. Tran, and R. Lbath, "Defining and using collaboration patterns for software process development," in *Proceedings of the 3rd International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2015) - CMDD*, 2015, pp. 557–564, doi: 10.5220/0005338705570564.
[7] Workflow patterns initiative site, Web: http://workflowpatterns.com.
[8] "Software & systems process engineering meta-model specification," OMG, Needham, formal/2008-04-01, 2008. [Online]. Available: https://www.omg.org/spec/SPEM/2.0/PDF.
[9] J. van Diggelen and M. Johnson, "Team design patterns," in *Proceedings of the 7th International Conference on Human-Agent Interaction*, Sep. 2019, pp. 118–126, doi: 10.1145/3349537.3351892.
[10] E. Gottesdiener, "Decide how to decide," *Software Development Magazine*, no. 1, 2001.
[11] U. Barchetti, C. Antonio, A. L. Guido, and L. Mainetti, "Modelling collaboration processes through design patterns," *Comput. Informatics*, vol. 30, no. 1, pp. 113–135, 2011, [Online]. Available: https://www.cai.sk/ojs/index.php/cai/article/view/156/132.
[12] A. Schmeil and M. J. Eppler, "Formalizing and promoting collaboration in 3d virtual environments – a blueprint for the creation of group interaction patterns," in *Facets of Virtual Environments. FaVE 2009*, vol. 33, F. Lehmann-Grube and J. Sablatnig, Eds. Berlin, Heidelberg: Springer, 2010, pp. 121–134.
[13] G. Eoyang, "Patterns of collaboration," 2018. Web: https://www.hsdinstitute.org/resources/patterns-of-collaboration.html (accessed Feb. 29, 2024).
[14] G. J. De Vreede, G. L. Kolfschoten, and R. O. Briggs, "ThinkLets: a collaboration engineering pattern language," *Int. J. Comput. Appl.*

Technol., vol. 25, no. 2/3, pp. 140–154, 2006, doi: 10.1504/IJCAT.2006.009064.

[15] A. V. Deokar, G. L. Kolfschoten, and G.-J. de Vreede, "Prescriptive workflow design for collaboration-intensive processes using the collaboration engineering approach," *Glob. J. Flex. Syst. Manag.*, vol. 9, no. 4, pp. 11–20, Oct. 2008, doi: 10.1007/BF03396547.

[16] A. Toniolo *et al.*, "Human-machine collaboration in intelligence analysis: An expert evaluation," *Intell. Syst. with Appl.*, vol. 17, p. 200151, Feb. 2023, doi: 10.1016/j.iswa.2022.200151.

[17] A. de Moor, "Community memory activation with collaboration patterns," in *Proceedings of the 3rd Prato International Community Informatics Conference (CIRN 2006)*, 2006, File: 2006\prato2006\demoorfinal.zip, 1 CD.

[18] J. F. Sowa, *Conceptual structures: information processing in mind and machine*, 1st ed. Boston: Addison-Wesley, 1983.

[19] A. de Moor, "Creativity meets rationale: collaboration patterns for social innovation," in *Creativity and Rationale*, vol. 20, J. Carroll, Ed. London: Springer, 2013, pp. 377–404.

[20] G. Falkman, M. Gustafsson, O. Torgersson, and M. Jontell, "The origin, representation, and use of collaboration patterns in a medical community of practice," in *Emerging Technologies and Information Systems for the Knowledge Society*, vol. 5288, M. D. Lytras, J. M. Carroll, E. Damiani, and R. D. Tennyson, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 403–412.

[21] W3C official website, "OWL 2 Web ontology language document overview," *W3C Recommendation*, 2012. Web: https://www.w3.org/TR/owl2-overview/.

[22] C. Dorn, G. Edwards, and N. Medvidovic, "Analyzing design tradeoffs in large-scale socio-technical systems through simulation of dynamic collaboration patterns," in *On the Move to Meaningful Internet Systems: OTM 2012.*, vol. 7565, R. Meersman, H. Panetto, T. Dillon, S. Rinderle-Ma, P. Dadam, X. Zhou, S. Pearson, A. Ferscha, S. Bergamaschi, and I. F. Cruz, Eds. Berlin, Heidelberg: Springer, 2012, pp. 362–379.

[23] C. Dorn and R. N. Taylor, "Architecture-driven modeling of adaptive collaboration structures in large-scale social web applications," Irvine, UCI-ISR-12-5, 2012. [Online]. Available: http://www.isr.uci.edu/tech_reports/UCI-ISR-12-5.pdf.

[24] H. Simon, "Rational decision making in business organizations," *Am. Econ. Assoc.*, vol. 69, no. 4, pp. 493–513, 1979, [Online]. Available: http://www.jstor.org/stable/1808698%0D.

[25] A. Smirnov, A. Ponomarev, T. Levashova, and N. Shilov, "Conceptual framework of a human-machine collective intelligence environment for decision support," *Proc. Bulg. Acad. Sci.*, vol. 75, no. 1, pp. 102–109, Feb. 2022, doi: 10.7546/CRABS.2022.01.12.