Modular Exoskeleton Design: Requirement Engineering with KAOS

Conference Paper · June 2016

DOI: 10.1109/BIOROB.2016.7523756

4 authors:

Rafael Sanchez Souza
University of São Paulo

Filippo Sanfilippo
Universitetet i Agder

José Reinaldo Silva
University of São Paulo

Arturo Forner-Cordero
University of São Paulo

Modular Exoskeleton Design: Requirement Engineering with KAOS*

Rafael Sanchez Souza[1], Filippo Sanfilippo[2], José Reinaldo Silva[3] and Arturo Forner Cordero[4]

***Abstract*— Wearable robots, such as exoskeletons, interact closely with the wearer. To this end, mechanical features are combined with control techniques to transparently follow human movements. Such expected behavior depends on specific project requirements, being safety the most critical one. According to the literature, modular robot design approaches provide flexible and yet robust solutions that meet stakeholder requirements. From this perspective, Goal Oriented Requirement Engineering (GORE) has been used as a tool for different systems; however, it has been more widely adopted in software rather than in hardware engineering.**

In this paper, GORE is adopted, with the KAOS tool, to fully exploit the integrated design of a modular exoskeleton - an adaptive mechatronic system. The balance of requirements with user safety constraints are analysed to advance in the project initial steps. It is shown that, although requirement modelling requires of an initial effort from the designer to formulate the goals, the proposed approach provides a more comprehensive system overview and documentation. Finally, the adoption of a semi-formal language justifies why a modular exoskeleton is a good choice when the design aims at meeting stakeholder requirements and improving user experience.

***Index Terms*— Modular Exoskeleton Design, Requirement Engineering, GORE, KAOS.**

I. Introduction

The development of new technologies has steered the design of complex and advanced robotic systems, such as exoskeletons, to achieve a high level of performance at reasonable production costs. Therefore, the design and construction of exoskeletons for rehabilitation, functional assistance or to improve the quality of life has become feasible [1]. The research and development of such kind of robots binds technology and science with a common goal: to produce a robust, safe, comfortable and economical assistive devices inspired by Nature. Nevertheless, unlike many other engineering products, there is no systematic and commonly established procedure to design exoskeletons.

There are major challenges involved in exoskeleton design. One of the most demanding tasks is that the system must be able to reproduce human movements to a functional level. This means more than 7 degrees of freedom to be controlled for upper limbs and up to 12 degrees of freedom for lower limbs [2]. In addition, it must be able to intervene in the exactly desired way with the subjects movement - to control the interaction force between user and robot. Many groups have been researching these technologies and some solutions have been proposed and implemented, such as the exoskeleton ARMin II [3] and the Impedance Control [4]. Yet, systems have functional - as the aforementioned - and non-functional requirements. Although many works discuss wearability, usability and cost [5], non-functional requirements are still often disregarded during technology development, as also is the case with exoskeleton research. Therefore, non-functional requirements, usually aimed at improve user acceptance represent a key factor in the decision-making process along the project development.

When designing such systems, modularity may be used as a possible approach. For instance, an upper limb exoskeleton prototype was developed in [6]. The mechanical design has a particular feature; the segments can be divided into similar modules formed by a joint and a link. This modular approach, while promising, has not been sufficiently explored in exoskeleton research [7]. Modularity brings development and maintenance costs down due to standardization of components, increasing portability, adaptability [8] and robustness [9]. Rapid prototyping may also be combined to obtain faster results [10], [11]. With modularity, non-functional requirements could also be implemented in an easier way.

However, a relevant drawback that affects some of the previous design approaches is that the Requirement Engineering (RE) phase has been frequently neglected and these works started directly in the modelling phase. In this perspective, our main focus is to evaluate the application of Goal Oriented Requirement Engineering (GORE), a goal-oriented approach belonging to the new generation of RE methods, to a modular exoskeleton design. Some benefits of the goal-oriented approach are: a more objective and effective initial design phase, a deeper analysis and specification of changes before costs get higher; the possibility of using different modelling languages; a better traceability (which could link further changes with resilient requirements), an opportunity to invest more time in project documentation; and the possibility of a better system monitoring and control and a better maintenance process.

There is a gap in the literature concerning formal exoskeleton design. Yet, while many works refer to "Exoskeleton Design", most start from the modelling phase, skipping requirements specification. Although RE has been proven to be very efficient in the design of complex software as well as

*This work was supported by CNPq and FAPESP (2010/17181-0)

[1]Rafael Sanchez Souza is with Department of Mechatronics and Mechanical Systems Engineering, Polytechnic School of University of Sao Paulo, Sao Paulo, Brazil. rafaelsouza@usp.br

[2]Filippo Sanfilippo is with the Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway. filippo.sanfilippo@ntnu.no

[3]José Reinaldo Silva is with Department of Mechatronics and Mechanical Systems Engineering, Polytechnic School of University of Sao Paulo, Sao Paulo, Brazil. reinaldo@usp.br

[4]Arturo Forner Cordero is with Department of Mechatronics and Mechanical Systems Engineering, Polytechnic School of University of Sao Paulo, Sao Paulo, Brazil. aforner@usp.br

mechanical devices, it is sometimes criticised for its practical effectiveness as it can be time-consuming, bureaucratic [12] and the benefits of using it are hard to measure. In addition, this method has been employed for an integrated design of both software and hardware, but only in few mechatronics design cases.

The goal of this paper is to present the first steps of a systematic exoskeleton design lifecycle and evaluate the method from a qualitative point of view. While some authors have already presented interesting formal modular robotic designs [9], [13], so far this method has not been applied to exoskeletons. In this paper, we show how to depart from high level requirements - goals - formulated in abstract diagrammatic language and to map them to a low level requirements - as technical as possible suited to a modular exoskeleton project. More effort is placed on the requirements phase with the use of the KAOS diagrammatic language (Fig.1 which can lead to formal requirements representation, expecting it to be as effective with a mechatronic system as with complex software [14] or other mechatronic artefacts [15].

The paper is organised as follows. The methodology is described in Section II. Successively, the selected system is described in Section III. The requirement modelling phase is outlined in Section IV. A discussion on the proposed method is provided in Section V. Finally, conclusions and future work are outlined in In Section VI.

II. Methodology

In this Section, we first shortly introduce some basic concepts of the adopted methodology. Then, the selected Goal Oriented Requirement Engineering (GORE) tool is described. Successively, the KAOS method is presented in its fundamentals. Systems Design has become the basis for designing any device, integrating product-centred approaches and services or process-approaches in what is known as a systemic view. Therefore, any device intended to have an operational coupling with a final user (a characteristic from "services") has to face a design process in which requirements are modelled and analysed beforehand. If hardware and software have to be integrated, modern approaches based on Model Driven Engineering and SysML modelling language could be used to combine static object-oriented design with process design. This process could be based on Petri Nets or other schematic formal representation. A complex product such as an exoskeleton should be considered a cluster of sub-systems responsible for specific behaviour which contributes to an overall goal.

User and exoskeleton are understood herein as a system composed of a set of agents with a collective goal extracted and analysed in the requirements phase. To do this, we propose a design lifecycle which is more concerned with single phases than with rationales and superposition as preconised by Rational Unified Process [17]. The considered phases are listed in the following:

- high-quality requirements elicitation;
- requirement modelling and analysis with traceability;
- solving conflicts and ambiguous requirements;
- decision process (for solutions) and Rationale documentation;
- design validation and verification.

A. Goal Oriented Requirement Engineering

In the literature, exoskeleton design usually starts with the modelling phase which is a normal application domain familiar to the design. As a consequence, the requirement phase is reduced, therefore potentially leading to a negative impact on the integration between the system and the final user.

Classic requirement analysis procedures have to cope with the dichotomy between functional and non-functional requirements, which makes the process less intuitive. On the contrary, in a goal-oriented approach, we do not have the same problem since goals already encompass all the necessary to achieve them including non-functional conditions. As stated in [18], "A goal is an objective the system under consideration should achieve". Goals are the fundamental stone on which the method elements such as objects, agents, events etc., will be supported. Some important characteristics of well defined goals are listed in the following:

- goals provide precise criterion for completeness of a requirement specification;
- goals provide precise criterion for rem Goals provide the are providedrationale for requirements;
- goal refinement is a natural mechanism for structuring complex requirements documents.

B. KAOS

KAOS is a representation schema to implement GORE [19] that is based on visual diagrams. These diagrams could also be transformed - once requirements compose a stable model - in a formal representation in LTL (Linear Tree Logic) or in Petri Nets [20]. The graphical user interface for KAOS implementation adopted herein is the Objectiver software, developed by Respect-IT. The KAOS framework is based on four linked diagrams that provide both designer and stakeholder a wider system overview of goals and operationalization. These four diagrams are briefly explained below (see Fig.1):

1) Goals Diagram. It is the fundamental model for KAOS in which the project goals are presented. Goals are refined into subgoals. The Goal Avoid[TissueDamageByExcessiveRangeOfMotion] may be refined to the Requirement Maintain[SafeRangeOfMotion] which is constrained by the Domain Property "Human Joint Safe Range of Motion" and again refined into the Requirement Maintain[AccurateJointAngleMeasure].
2) Object Model. It allows for the identification of objects, such as entity, relationship, event or agent. Defines the concepts of the application domain and the system constraints. It also allows for establishing the object characteristics as attributes. The Exoskeleton Joint and the Human Joint relation are defined as "Attached".

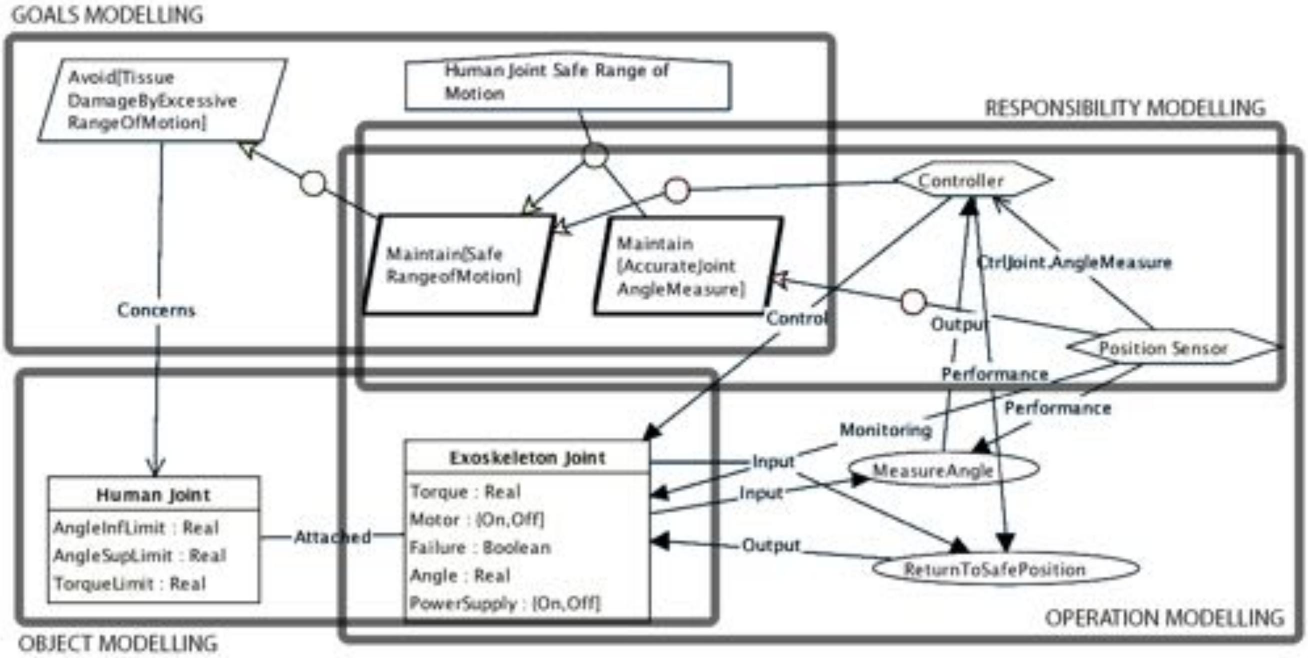

Fig. 1. The KAOS framework adapted from [16].

3) Agent Responsibility Model. It defines the responsibilities between agents and requirements. The Controller and the Position Sensor are responsible for maintaining a safe range of motion and an accurate joint angle measure.
4) Operation Model. It shows how the objects work together to achieve the system requirements by defining the state transitions. If angles are off limits, the Controller will bring the Exoskeleton Joint back to a safe position.

III. System Description

A typical Systems Design approach considers the known system as-is, that is, the legacy system from where the knowledge to carry out the design is taken. As presented in Fig.2, the proposed device - the modular exoskeleton Mod-Exo on the right side of the figure - is considered the system-to-be. We consider the EXO-C [21] prototype (Fig.2, as the legacy system.

In [6]], a mechanical structure of an exoskeleton, including motor and reduction, was built. This exoskeleton can be divided into modules and it's referred here as Mod-Exo. Modularity is a property that can also be applied to both software and hardware and KAOS is expected to help this integrated design.

The EXO-C actuation is provided by Maxon Motor EC32, 80 Watt with a 18:1 planetary reduction acting on a ball-screw axis and driven by a Maxon EPOS2 24/5 driver. The interaction force between the arm and the exoskeleton is obtained by means of a Wheatstone bridge of 350 ohm strain gauges glued to the exoskeleton aluminium structure attached to the forearm. The position is measured by a

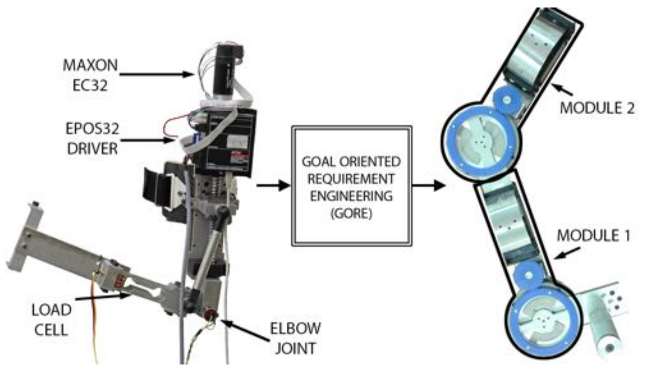

Fig. 2. The underlying idea of applying Goal Oriented Requirement Engineering (GORE), a software design tool borrowed from software engineering, for the integrated hardware and software design of a modular exoskeleton. In the figure the EXO-C (the legacy system) and the Mod-Exo (modular exoskeleton design).

rotational potentiometer 2 Kohm with 2 percent linearity coupled to the elbow joint. There are two levels of control, a lower level one provided by the driver EPOS2 24/5 from Maxon and a higher level implemented in an embedded computer PC104. There are different high level controllers that can be easily implemented. So far, a force-follower and an impedance controller have been implemented. The exoskeleton is modular and can be anchored to the arm with a standard commercial orthosis, such as the shoulder orthosis Ottobock Omo-Neurexa (Otto Bock gmBh, Germany).

IV. Requirement Modelling

Requirement Modelling occurs immediately after a primary goal specification by the stakeholder and before the system modelling and analysis. The underlying idea is to develop a diagram containing the project constraints, which should be as complete, precise and unambiguous as possible. The goal is to improve the quality of the information provided by the stakeholder and translate it into tangible information for the engineer. The target is to prevent the engineer from working on requirements that will not fit the stakeholders needs and to reduce the design iterations.

This section shows a brief implementation of the method. The steps followed were fully described in [22]. Note that this process is not done once only, but it is continuously repeated along the project development.

A. Identifying Preliminary Goals and Refinement

Preliminary goals were retrieved from the literature and from the previous research at the Biomechatronics Laboratory of the Escola Politecnica of the University of Sao Paulo. They were considered to elaborate the three subprojects main diagrams, represented in figures 3, 4, 5; the diagrams are self-explanatory, which makes them useful for documentation, as follows:

1) The system should intervene in user movement. This goal concerns the control of the exoskeleton, that is, the inputs and outputs to steer or modify the movement of the user. Also, this goal defined the joints and degrees of freedom required to perform the task;
2) The system must be commercially feasible. This goal targets an exoskeleton that goes beyond the academic environment. It gathers economical, environmental challenges and users acceptance - some taken from [5], such as comfort, costs and ease of use;
3) User safety. A primary concern of any wearable robot;
4) The system must be a modular robot. There is strong evidence that this property may be a key to satisfy the system requirements and it is considered here as a hypothesis. Modular preliminary subgoals were identified from [8].

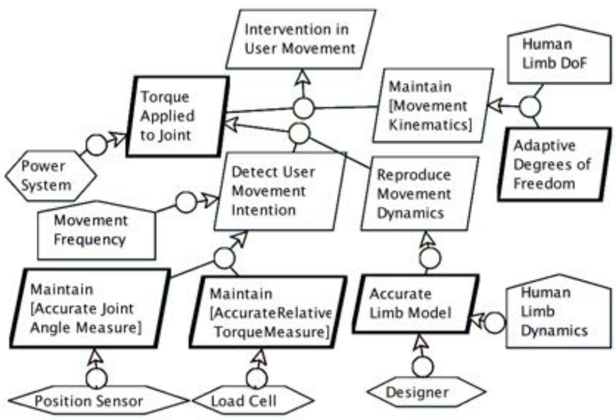

Fig. 3. Mod-Exo: User's Movement Intervention Goals.

B. Formalizing Goals and Identifying Objects

As a case study for the KAOS implementation, the next steps are followed towards the goal of Maintain User Safety, a primary concern of any exoskeleton. The Goals are first discursively defined, then formally defined and finally derived to objects.

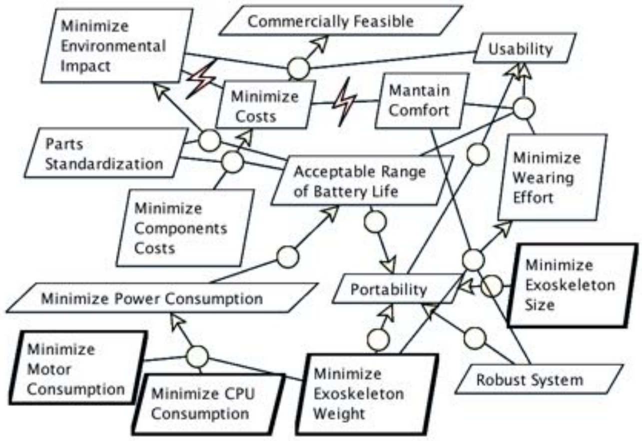

Fig. 4. Mod-Exo: Commercial Feasibility Goals.

Considering the subgoals Avoid [Muscle Bone Stress] and Avoid[Joint Hyper Extension Flexion]: both are simultaneously necessary for the goal Maintain [User Safety]. This is represented by a circle as an AND refinement in Fig.6. These subgoals can be refined into requirements, represented by the thick border parallelogram Maintain[Joint Torque Limit] and Maintain[Joint Angle Limit]. If the human joint torque limit is respected, the exoskeleton will not harm the user. Similarly, the same will happen if the human joint angles ranges of motion are respected. This is represented by the OR refinement at Maintain[Join Angle Limit] in Fig.6. In the following, goals are defined differently, in a formal language, which helps to identify project parameters.

Goal Maintain[SafeRelativeTorque]
Definition The Exoskeleton Joint torque should stay below the maximum torque the human joint can handle.
FormalDef

$$\forall ej : ExoJoint, hj : HumanJoint\ Attached(ej, hj) \Rightarrow ej.Torque \leq hj.SafeTorque. \tag{1}$$

Goal Maintain[SafeRangeOfMotion]
Definition The Exoskeleton Joint angle should stay between the maximum and minimum angles the human joint can handle.
FormalDef

$$\forall ej : ExoJoint, hj : HumanJoint\ Attached(ej, hj) \Rightarrow hj.AngleInfLimit \leq ej.Angle \leq hj.AngleSupLimit. \tag{2}$$

From the formal definition, objects and attributes can be identified as seen in Fig. 6: ExoJoint and HumanJoint are the objects corresponding to the exoskeleton module and the user arm; "Attached" defines the relationship between them; ej.Torque and hj.SafeTorque are declared as their respective attributes.

C. Further Eliciting Goals Through WHY and HOW Questions

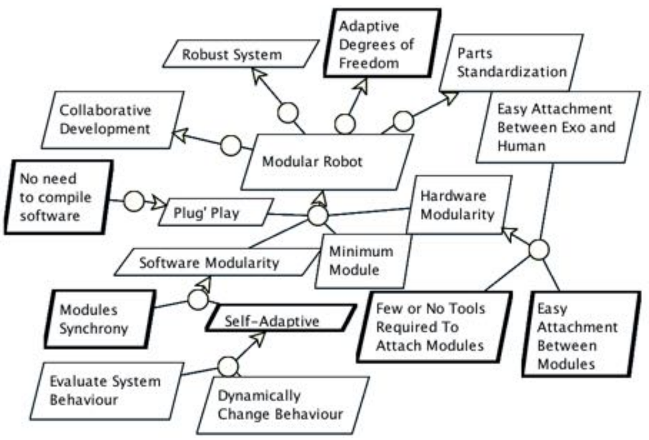

Fig. 5. Mod-Exo: Modularity Goals.

WHY and HOW questions are asked at this step to elicit further goals. WHY identifies higher-level parent goals that provide the rationale for the project documentation. HOW refines a goal into subgoals until they can be assigned to agents. By asking HOW in Maintain[SafeRangeOfMotion], a redundant security solution is also defined: in the case of a software failure, the power supply should be switched off.

Goal Maintain[SystemShutdownWhenAngleLimit]
Definition Electrically constraints operational angles with sensors attached directly to power source.
FormalDef

$$\begin{aligned}&\forall ej : ExoJoint, hj : HumanJoint : \\ &(hj.AngleInfLimit \leq ej.Angle \leq hj.AngleSupLimit) \wedge \\ &\qquad Attached(ej, hj) \Rightarrow ej.Power = \text{“}Off''.\end{aligned} \tag{3}$$

By asking the Modular Robot Goal WHY, the higher-level goals: Robust System, Parts Standardization, Minimise Wearing Effort, Portability, Adaptive Degrees of Freedom, Minimise Costs, Collaborative Development are identified, completing the diagram shown in Fig. 5.

In Fig. 6, “Human Joint Safe Working Torque” and “Human Joint Safe Rage of Motion”, inside the pentagons are Domain Properties. These are properties related to the environment such as laws of nature.

D. Identifying potential responsibility assignments

The agents identified through the previous steps are: Load Cell and Position Sensor, responsible for monitoring the system; the Controller, which centralizes information, processes it and outputs commands; and the Micro Switch, an electric switch that should be positioned according to range of motion requirements. There are goals directly related to these agents, as listed in the following:

Goal Maintain[AccurateJointAngleMeasure]
Definition The angle measure should equal the actual angle of the exoskeleton joint.
FormalDef

$$\begin{aligned}&\forall ej : ExoJoint, hj : HumanJoint, c : Controller \\ &\quad CtrlJoint(c, ej) \wedge Attached(ej, hj) \Rightarrow \\ &\qquad c.AngleMeasure = ej.Angle.\end{aligned} \tag{4}$$

Goal Maintain[AccurateRelativeTorqueMeasure]
Definition The torque measure should equal the actual torque between user and exoskeleton.
FormalDef

$$\begin{aligned}&\forall ej : ExoJoint, hj : HumanJoint, c : Controller \\ &\quad CtrlJoint(c, ej) \wedge Attached(ej, hj) \Rightarrow \\ &\qquad c.TorqueMeasure = ej.Torque.\end{aligned} \tag{5}$$

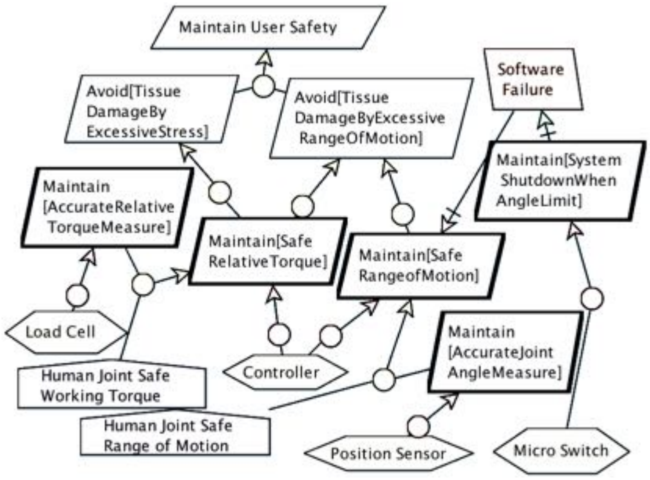

Fig. 6. Further Eliciting Safety Goals.

V. Discussion

We have applied a formal and structured design method to define the requirements of an exoskeleton. One of the questions was if the use of this approach would really be advantageous, considering the concerns about its effectiveness. The learning curve is slightly steep from the designer point of view, since there are several aspects to master before a proper model is achieved. The initial effort is placed on the formulation and refinement of the goals, since a bad goal definition may lead to a dead end. Due to the uncommon application of the KAOS method to mechatronic systems, there are few references in literature to rely on when struggling with a procedural concept: sensors, for example, in this paper were defined as agents but they could also be entities.

According to [23], the KAOS method evaluation should take Requirement Engineering Objectives, such as Pertinence, Correctness, Traceability and Understandability, into consideration.

The models for each defined goal came out differently. The “Maintain User Safety” model can be considered the most successful; with the refinements, it is possible to reach tangible project requirements departing from an abstract “Maintain User Safety” and even think of safety solutions, such as the micro switch, conceived during the development of the models. “Intervention in User Movement” may be also considered quite successful, but it still needs further development. We believe that the differences in complexity of each goal are directly related to the model effectiveness. This was verified in the two models, where the first is simpler then the second.

Although not so straightforward, both “Commercially Feasible” and “Modular Robot” goals may provide some interesting results for the exoskeleton designer. In the beginning of the study, the modularity goal was stated as a possible solution to meet the exoskeleton design requirements. These statements are observed to be met, when asking HOW to commercially feasibility and WHY to modularity. Also, the complexity of the diagrams indicates the challenge of producing a commercially feasible exoskeleton: a solution that besides solving control and kinematic challenges, meets users needs.

The models highlight important project parameters such as “Human Joint Safe Working Torque” and “Human Joint Safe Rage of Motion” which must be investigated before the exoskeleton prototype is built. The bolts linking “Minimise Costs” and “Minimise Environmental Impact” and “Maintain Comfort” in Fig.4 indicate conflicts between the goals. The designer has to reflect upon these indicatives given by the models.

Although connected to requirement elicitation, which is not formal, requirements analysis using the goal-oriented approach and the KAOS language may result in a formal description in LTL (Linear Tree Logic) or in Petri Nets to cover the dynamics of the system [24].

VI. Conclusion

In this paper, it has been shown that the adoption of the goal-oriented approach for requirement engineering may be used to enrich the early phase of exoskeleton design because it highlights constraints, inserts traceability into the process, and can also result in a formal specification as the first goal in the classic design process.

To the best of our knowledge, there are only a few papers that addressed the problem of exoskeleton requirements definition. Some works, such as [5], [25] raised requirements by means of a survey. We believe that these procedures are complementary to the work presented here: KAOS provides tools to evaluate those requirements pertinence and to further develop those requirements.

With respect to the modelling, a drawback may be that the refinement time grows exponentially in order to cover all the system features. It is strongly suggested that this method is used objectively and according to predefined milestones. Once a certain requirement modelling time is reached, the designer should move forward to the next design phases and update the models along the project development.

The next steps should be to advance to the next design phases, which comprehend modelling and implementation of the solutions that meet the requirements raised by the KAOS method. Other research groups are encouraged to try the method and to improve the models presented herein.

References

[1] A. Ruiz, A. Forner-Cordero, E. Rocon, and J. Pons, “Exoskeletons for rehabilitation and motor control,” in *Proc. of the 1st IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Pisa, Italy*, 2006, pp. 601–606.

[2] J. C. Perry and J. Rosen, “Design of a 7 degree-of-freedom upper-limb powered exoskeleton,” in *Proc. of the 1st IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Pisa, Italy*, 2006, pp. 805–810.

[3] M. Mihelj, T. Nef, and R. Riener, “Armin ii-7 dof rehabilitation robot: mechanics and kinematics,” in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA), Roma, Italy*, 2007, pp. 4120–4125.

[4] N. Hogan, “Impedance control: An approach to manipulation,” in *Proc. of the IEEE American Control Conference*, 1984, pp. 304–313.

[5] J. Wolff, C. Parker, J. Borisoff, W. B. Mortenson, and J. Mattie, “A survey of stakeholder perspectives on exoskeleton technology,” *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, p. 169, 2014.

[6] A. B. W. Miranda, “Exoesqueleto robótico de membro superior com três graus de liberdade ativos,” 2014.

[7] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy, and S. Leonhardt, “A survey on robotic devices for upper limb rehabilitation,” *Journal of NeuroEngineering and Rehabilitation*, vol. 11, no. 3, pp. 10–1186, 2014.

[8] T. Matsumaru, “Design and control of the modular robot system: Tomms,” in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA), Nagoya, Aichi, Japlan*, vol. 2, 1995, pp. 2125–2131.

[9] J. Fryer, G. McKee, P. S. Schenker *et al.*, “Configuring robots from modules: an object oriented approach,” in *Proc. of the 8th IEEE International Conference on Advanced Robotics (ICAR), Monterey, California, U.S.A.*, 1997, pp. 907–912.

[10] F. Sanfilippo, H. Zhang, K. Y. Pettersen, G. Salvietti, and D. Prattichizzo, “ModGrasp: An open-source rapid-prototyping framework for designing low-cost sensorised modular hands,” in *Proc. of the 5th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), São Paulo, Brazil*, 2014, pp. 951–957.

[11] F. Sanfilippo, H. Zhang, and K. Y. Pettersen, “The new architecture of ModGrasp for mind-controlled low-cost sensorised modular hands,” in *Proc. of the IEEE International Conference on Industrial Technology (ICIT2015), Seville, Spain*, 2015, pp. 524–529.

[12] B. Nuseibeh and S. Easterbrook, “Requirements engineering: a roadmap,” in *Proc. of the ACM Conference on the Future of Software Engineering*, 2000, pp. 35–46.

[13] Z. Bi and W.-J. Zhang, “Concurrent optimal design of modular robotic configuration,” *Journal of Robotic systems*, vol. 18, no. 2, pp. 77–87, 2001.

[14] V. M. B. Werneck, A. d. P. A. Oliveira, and J. C. S. do Prado Leite, “Comparing gore frameworks: i-star and kaos.” in *WER*, 2009.

[15] K. Jansckek, *Mechatronic Systems Design*. Springer, 2012.

[16] I. Respect, “A kaos tutorial,” 2007.

[17] P. Kruchten, *The Rational Unified Process: An Introduction*. Addison-Wesley, 2004.

[18] A. Van Lamsweerde, “Goal-oriented requirements engineering: A guided tour,” in *Proc. of the 5th IEEE International Symposium on Requirements Engineering*, 2001, pp. 249–262.

[19] W. Heaven and A. Finkelstein, “Uml profile to support requirements engineering with kaos,” *IEEE Proceedings-Software*, vol. 151, no. 1, pp. 10–27, 2004.

[20] J. M. Silva and J. R. Silva, “Using goal-oriented requirement analysis in the design of product-service artifacts,” in *Proc. of the XII SBA Brazilian Symposium in Automation*, 2015, pp. 1601–1606.

[21] A. B. W. Miranda, A. Y. Yasutomi, C. Souit, and A. Forner-Cordero, “Bioinspired mechanical design of an upper limb exoskeleton for rehabilitation and motor control assessment,” in *Proc. of the 5th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Rome, Italy*, 2012, pp. 1776–1781.

[22] E. Letier *et al.*, “Reasoning about agents in goal-oriented requirements engineering,” Ph.D. dissertation, PhD thesis, Université catholique de Louvain, 2001.

[23] H. S. Al-Subaie and T. S. Maibaum, "Evaluating the effectiveness of a goal-oriented requirements engineering method," in *Proc. of the 4th IEEE International Workshop on Comparative Evaluation in Requirements Engineering (CERE), Minneapolis/St. Paul, Minnesota, U.S.A.*, 2006, pp. 8–19.
[24] J. M. Silva and J. R. Silva, "Combining kaos and ghenesys in the requirement and analysis of service manufacturing," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 1634–1639, 2015.
[25] V. Patoglu, G. Ertek, O. Oz, D. Zoroglu, and G. Kremer, "Design requirements for a tendon rehabilitation robot: results from a survey of engineers and health professionals," in *Proc. of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2010, pp. 85–94.