Chapter 13

Allowing a Real Collaboration Between Humans and Robots

Andrea Casalino

13.1 Introduction

Up to the recent past, the paradigm universally adopted for industrial robotics provided for the strict segregation of robots in protected environments, adopting fences or optical barriers. Only recently, the potential benefits of a collaboration between humans and robots have gained the attention of roboticists [10], mainly motivated by the Industry 4.0 paradigm [15]. Since many tasks are still impossible to be fully automatized, it seems natural to let humans and robots cooperate: highly cognitive actions are undertaken by humans, while those requiring high precision and repeatability are performed by robots. A well established literature addressed the problem of a safe coexistence in a shared space by introducing motion control techniques [6], based on the use of sensors perceiving the scene and tracking the motion of the human operators. Corrective trajectories can be planned on-line with the aim of dodging the human and, at the same time, keep driving the manipulator to the desired target position.

Such initial works conceived the robots as something that should interfere as less as possible with the humans populating the same cell, only opening the door for a real collaboration. The possibility for a properly instrumented robotic device to understand and somehow predict humans' intentions is now considered as important as safety and it is possible by providing the robots with the proper cognitive capabilities. In this context, a crucial role is played by vision sensors since the analysis of human motion is one of the most important features. Many results have been reported showing the increasing capability of robots to semantically interpret their human fellows. In [11] a method based on conditional random files (CRF) is used by the robot to anticipate its assistance. In [12] Gaussian Mixture Models (GMMs) are used to predict human reaching targets.

A. Casalino (✉)

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Piazza Leonardo Da Vinci 32, Milan, Italy
e-mail: andrea.casalino@polimi.it

© The Author(s) 2021

A. Geraci (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-030-62476-7_13

Collaborative assemblies are typical applications of collaborative robotics. In such scenarios, the actions of the robots influence the ones of the humans, and the optimization of the robotic action sequence becomes crucial. Reference [4] describes a genetic algorithm for a collaborative assembly station which minimises the assembly time and costs. In [14], a trust-based dynamic subtask allocation strategy for manufacturing assembly processes has been presented. The method, which relies on a Model Predictive Control (MPC) scheme, accounts for human and robot performance levels, as well as for their bilateral trust dynamics. By taking inspiration from real-time processor scheduling policies, [9] developed a multi-agent task sequencer, where task specifications and constraints are solved using a MILP (Mixed Integer Linear Programming) algorithm, showing near-optimal task assignments and schedules. Finally, [13] proposes a task assignment method, based on the exploration of possible alternatives, that enables the dynamic scheduling of tasks to available resources between humans and robots. Many alternative plans are computed in [16], where a dynamic task allocation of activities is also possible, by taking into account the different capabilities of agents (robots and humans). In [3], the scheduling problem is solved using a Generalised Stochastic Petri Net as a modelling tool. The selection of the optimal plan takes into account the amount of time for which the agents remain inactive, waiting for the activation of some tasks.

All the aforementioned works seem to model the human operators as no more than highly cognitive manipulators, that can be instructed to do certain actions. In this work timed Petri nets will be adopted to model the human-robots multi agent system, however assuming the human as an uncontrollable agent, whose actions must be recognized in order to decide which complementary actions the robots should undertake. The remaining of this chapter is structured as follows: Sect. 13.2 will discuss the activity recognition problem, while Sect. 13.3 will present the approach followed to predict the future ones. Such predictions will be crucial for applying the scheduling approach described in Sect. 13.4.

13.2 Recognizing the Human Actions

Recognizing the actions performed by a human is easy and natural for another human being. It is done by mainly analysing the motion of the arms. This is the same approach followed by computer vision algorithms. In the following, an approach based on RGB-d cameras and factor graphs will be discussed. Assume the (finite) set of possible human actions $\mathcal{A} = \{a_1, \dots, a_m\}$ known. The aim of the algorithm proposed in this section is to detect the starting and the ending time of each action, by analysing the motion performed by the operator in the recent past. RGB-D sensors can be exploited to keep track of some points of interest in the human silhouette, see the top left corner of Fig. 13.1. The signals retrieved are subdivided into many sub-windows each having a maximum length of l_w samples. Then, a feature vector $F_O^i = [O_1^i \dots O_F^i]$ made of F components can be extracted from the i th window, representing an indirect indication of the action $a \in \mathcal{A}$ that was performed within

Fig. 13.1 A RGBd camera can be used to keep track of the skeletal points (cyan dots in the top left corner) and compute the skeletal distances over time. The segmenting graph is built considering a segmentation hypothesis ρ and the observations are partitioned accordingly

the same window. A good selection for the feature set is crucial and can be made with deep learning techniques for instance. Clearly, a specific action a must produce characteristic values for F_O . In the layout proposed in Sect. 13.4, the inter-skeletal distances as well as the distance of the operator's wrists to some particular positions of the space were empirically found to be effective.

The main problem to overcome when segmenting human actions is that the exact durations of the actions (and consequently the starting and ending time instants) as well as their total number is not precisely known. To handle this aspect, a probabilistic framework must be considered and the most probable sequence of actions that produced a certain macro-window of observations must be determined. This problem can be tackled considering factors graph [7], having as hidden variables the sequence of actions actually performed by the human and as evidences a battery of features $F_O^{1,\dots,L}$. Vector $\rho = [\rho_1 \dots \rho_S]$ is adopted for describing the durations of human actions as well as their number. Indeed, ρ_i indicates the percentage of time spent by the human when doing the i th action in the sequence. Knowing ρ , it is possible to build the underlying factor graph, assuming to connect $\rho_i \cdot L$ observations to the node representing the i th action, refer to Fig. 13.1. The potentials Ψ_{OA}^i must express the correlation existing among the features and the actions, while those correlating the actions over time, Ψ_{AA}^i , should take into account precedence constraints among the actions or any kind of prior knowledge about the process. Since the real segmentation ρ^* describing the sequence of actions is not available, the proposed

approach considers many hypotheses¹ $\rho^1, \rho^2, \dots$, which are iteratively compared with the aim of finding the optimal one $\hat{\rho}$, i.e. the one more in accordance with the observations retrieved from the sensors. More specifically, a genetic algorithm [8] can be efficiently adopted assuming as a fitness function $\mathcal{C}$ the following likelihood:

$$\mathcal{C}(\rho) = \mathcal{L}(\rho | O_1^1, \dots, O_F^L) = \mathbb{P}(O_1^1, \dots, O_F^L | \rho) \mathbb{P}(\rho)_{prior} \quad (13.1)$$

where $\mathbb{P}(O_1^1, \dots, O_F^L | \rho)$ can be computed by doing belief propagation on the factor graph pertaining to a specific hypothesis ρ . Determining the optimal segmentation $\hat{\rho}$ completes the segmentation task: the number of actions done by the operator is assumed equal to the number of segments in ρ and for the j th segment the marginal distribution $[\mathbb{P}(A_j = a_1 | O_1^1, \dots, O_F^L, \rho^*) \dots \mathbb{P}(A_j = a_m | O_1^1, \dots, O_F^L, \rho^*)]$ expresses in a probabilistic way which action was done within that segment.

13.3 Predicting the Human Actions

By knowing the sequence of actions done in the recent past, predictions about the future ones can be made. In particular, we are interested in evaluating the waiting time τ^a before seeing again a particular action $a \in \mathcal{A}$.

Human assembly sequences usually form quasi-repetitive patterns. In other words, the sequence of human activities can be modelled through a time series, which is the output of a certain dynamic process. Assume A_k as the ongoing activity at discrete time instant k , the behaviour of the human fellow co-worker can be modelled through the following discrete-time process:

$$\begin{aligned} A_{k+1} &= f(A_k, A_{k-1}, A_{k-2}, \dots, A_{k-n}) \\ t_{k+1} &= t_k + g(A_k) \end{aligned} \quad (13.2)$$

where $t_k \in \mathbb{R}^+ \cup \{0\}$ represents the time instant corresponding to the transition from A_{k-1} to A_k and $g(a) = T^a > 0$ is the duration of activity $a \in \mathcal{A}$. The stochasticity of the underlying discrete process governing the sequence of activities can be modelled by making use of a higher order Markov Model [5], which computes the predicting probability distribution associated to the next activity in the following way:

$$\begin{aligned} \mathbb{P}(A_{k+1} = a | A_k = k_0, \dots, A_{k-n} = k_n) &\approx \\ &\approx \sum_{i=0}^n \lambda_i \mathbb{P}(A_{k+1} = a | A_{k-i} = k_i) \end{aligned} \quad (13.3)$$

¹Notice that also the number of elements in ρ is not known and many hypothesis with a different number of actions should be considered.

hence a mixture model that corresponds to usual Markov Chains for $n = 0$ and requires only $m^2(n + 1)$ parameters. A prediction of the probability distribution $\hat{X}_{k+1}$ at time $k + 1$ can be computed as

$$\hat{X}_{k+1} = \sum_{i=0}^n \lambda_i Q_i X_{k-i} \quad (13.4)$$

where $X_{k+1} = [\mathbb{P}(A_{k+1} = a_1) \dots \mathbb{P}(A_{k+1} = a_m)]^T$ refers to the probability distribution describing the state of the system² at step $k + 1$, while Q_i denotes the i -steps transition probability matrix that can be simply evaluated through counting statistics. The weights λ_i are estimated on-line, by minimizing the prediction errors on a window of n recent observations:

$$\min \left(\left\| \sum_{i=0}^n \lambda_i Q_i X_{k-i} - X_{k+1} \right\|^2 \right) \quad (13.5)$$

It is not difficult to show that this leads to the solution of a quadratic programming problem in the following form:

$$\min_{\lambda} \|A\lambda - b\|^2 \text{ subject to } \sum_{i=0}^n \lambda_i = 1, \text{ and } \lambda_i \geq 0 \quad (13.6)$$

Assume that the duration of an activity $a \in \mathcal{A}$, i.e. T^a , can be modelled as a stochastic variable with a strictly positive lower bound, i.e. $T^a \geq T_{min}^a > 0$. In order to estimate the waiting time needed for the certain activity a to show up, say τ^a , we can combine this information with the model describing the activity sequence, Eq. (13.3). In particular, at the present continuous time instant $\bar{t}$, given the sequence of the last activities (possibly including the currently running one) $A_k, A_{k-1}, \dots, A_{k-n}$, we would like to estimate the probability distribution of the waiting time for the beginning of a certain activity a , i.e. $\mathbb{P}(\tau^a \leq t | A_k, A_{k-1}, \dots, A_{k-n})$. The key idea is to construct a predictive reachability tree. Then, evaluating the time spent to traverse each possible branch in the tree, terminating with the desired activity $a \in \mathcal{A}$, it is possible to estimate τ^a . Since the reachability tree is, in principle, infinite, a prediction horizon ΔT must be defined, meaning that the given probability will be computed up to the instant $t = \bar{t} + \Delta T$. The probability associated to each branch can be simply computed by multiplying the probability of each arc of the branch, i.e. $p_{branch} = \prod_{(i,j) \in branch} p(i,j)$. As for the waiting time associated to each branch τ_{branch} , this is simply the sum of the duration of each activity T^a , i.e. $\tau_{branch} = \sum_{j:(i,j) \in branch} T^j$. The time associated to each branch is computed as the sum of stochastic variables which are empirically approximated using the statistics associated to recently acquired samples, using a Monte Carlo numerical approach.

²Which is exactly the output of the algorithm described in Sect. 13.2.

Finally, given the distributions of the times associated to each branch, the overall distribution of the waiting time of the activity a can be simply computed as a weighted sum of the waiting times associated to each branch, i.e.

$$\begin{aligned} \mathbb{P}(\tau^a \leq t | A_k, A_{k-1}, \dots, A_{k-n}) &= \\ &= \sum_{branch} p_{branch} \mathbb{P}(\tau_{branch} \leq t). \end{aligned} \quad (13.7)$$

13.4 Assistive Scheduling

Detecting and predicting the human actions is crucial when planning the robotic ones. Assuming the tasks as pre-assigned to all the agents (robots and humans), the aim of scheduling become essentially to control the robots in a way as much as possible compliant with the human (predicted) intentions. To this purpose the cell is modelled as a multi agent system by making use of timed Petri Nets [1]. In particular, the robots will be modelled as usual controllable agents, while humans will be assumed as uncontrollable, even if they will affect the system when doing the assigned tasks. Modelling Timed Petri Nets can be done by following systematic rules, from a description of the tasks assigned to both humans and robots as well as their precedence constraints, Figs. 13.2 and 13.3. The robotic tasks are modelled in a canonical way with two transitions: the first starting the task and the second terminating it. For every robots, an additional action is always considered and consists of an idle of a quantum of time (orange transitions in Fig. 13.3), able to postpone the starting of the following action (multiple idle are also possible). The human actions are modelled with three transitions: the first one firing when the human is predicted to be ready to start the corresponding action (also indicated as intentional transition), the second one firing as soon as the human actually starts that action and the last one terminating it. Red places in Fig. 13.3 are those related to idle states for the agents.

Scheduling essentially consists in deciding the commands to dispatch to the robots, or in other words resolving the controllable conflicts. To this purpose, a model pre-

Fig. 13.2 Example of task allocation of some assembly sequences

Fig. 13.3 The timed Petri nets adopted to model the co-assemblies reported in Fig. 13.2. The global net modelling the system can be obtained by superimposing the reported three ones. The transitions reported in yellow are those assumed as controllable. Orange transitions are controllable too and correspond to idling of a quantum of time. Red places are those related to idle state for an agent of the system

Fig. 13.4 The assistive scheduling approach

dictive approach is exploited and the underlying Petri Net is used to compare possible future evolutions of the systems with the aim of finding those minimizing the inactivity times and indirectly maximizing the throughput. Since the system is only partially controllable, a stochastic approach is assumed and the optimal policy is recomputed every time by exploring the reachability tree of the net with a Monte Carlo simulation, details are explained in [2]. The predictions about the future human activities, Sect. 13.3, are used to build the distributions modelling the intentional transitions. The pipeline of Fig. 13.4 summarizes the entire approach.

13.5 Results

The assistive scheduling approach described so far was applied in a realistic co-assembly, whose workspace consisted in: one IRB140 ABB robot, the collaborative dual-arm robot YuMi of ABB, two automatic positioners (treated like additional robots to control) and a MICROSOFT KINECT v2 as monitoring device, see Fig. 13.5. The aim of the collaboration was to perform the assembly of a box containing a USB pen.³ 20 volunteers were enrolled for the experiments and were divided into two equally sized groups named Group 1.A and Group 1.B. Operators in Group 1.A performed the assembly while the strategy discussed in Sect. 13.4 was adopted.

³The assembly steps are described at https://youtu.be/_Jxo1mNZ1C8.

Fig. 13.5 The experimental setup with the two robots, the carts and the human operator

Instead, for those in Group 1.B a centralized strategy was adopted for deciding the actions of all agents, treating the humans like additional robots to control.

The pictures at the top of Fig. 13.6 report the inactivity times measured during the experiments. As can be seen, the assistive approach applied for Group 1.A is able to significantly reduce waste of times. This reflects in an improved throughput of the system as can be appreciated in the bottom left corner picture of Fig. 13.6. It is worth to recall that the approach followed to predict the human intentions, Sect. 13.3 is data-driven, adapting to the human over time. This fact gives the scheduler some learning capabilities or in other words the produced plans are able to constantly improve the way the robots are controlled, refer to the bottom right corner of Fig. 13.6.

13.6 Conclusions

This work aimed to study how to allow humans and robots to actively collaborate for performing a shared tasks, like the ones involved in co-assemblies. Humans were not conceived as additional controllable agents which are instructed to perform actions scheduled by a centralized planner. Instead, the decision-making capabilities of the operators were exploited, since they were allowed to drive the evolution of the plant. In this context, the robotic mates were endowed with the cognitive capabilities required for both recognizing and predicting the human actions. Regarding action recognition, particular attention was paid to manage the fact that the starting and ending time instants of the action were not known. To overcome such issue, genetic algorithms were deployed in conjunction with factor graphs. For what concerns the prediction problem, a data-driven approach was described, using an adaptive Higher order Markov model to forecast the sequence of human actions. The scheduling of the robotic actions was done with a kind of scenario based approach, which needs to explore the reachability tree of a timed Petri Net modelling the agents in the robotic cell.

Fig. 13.6 Results showing the benefits of an assistive scheduling, Group 1.A w.r.t to a centralized approach, Group 1.B

References

1. Berthomieu B, Diaz M (1991) Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans Softw Eng* 17(3):259–273
2. Casalino A, Zanchettin A, Piroddi L, Rocco P (2019) Optimal scheduling of human-robot collaborative assembly operations with time Petri nets. *IEEE Trans Autom Sci Eng*
3. Chen F, Sekiyama K, Huang J, Sun B, Sasaki H, Fukuda T (2011) An assembly strategy scheduling method for human and robot coordinated cell manufacturing. *Int J Intell Comput Cybern* 4:487–510
4. Chen F, Sekiyama K, Cannella F, Fukuda T (2014) Optimal subtask allocation for human and robot collaboration within hybrid assembly system. *IEEE Trans Autom Sci Eng* 11(4):1065–1075
5. Ching WK, Fung ES, Ng MK (2004) Higher-order Markov chain models for categorical data sequences. *Nav Res Logist (NRL)* 51(4):557–574
6. Flacco F, Kroger T, De Luca A, Khatib O (2012) A depth space approach to human-robot collision avoidance. In: 2012 IEEE international conference on robotics and automation, pp 338–345

7. Frey BJ (2002) Extending factor graphs so as to unify directed and undirected graphical models. In: Proceedings of the nineteenth conference on uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., pp 257–264
8. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning, Addison Wesley, Reading, MA. The applications of GA-genetic algorithm for dealing with some optimal calculations in economics
9. Gombolay MC, Wilcox R, Shah JA (2013) Fast scheduling of multi-robot teams with temporospatial constraints. *Robotics: science and systems*
10. Haddadin S, Suppa M, Fuchs S, Bodenmüller T, Albu-Schäffer A, Hirzinger G (2011) Towards the robotic co-worker. Springer, Berlin, pp 261–282
11. Koppula HS, Saxena A (2016) Anticipating human activities using object affordances for reactive robotic response. *IEEE Trans Pattern Anal Mach Intell* 38(1):14–29
12. Luo R, Hayne R, Berenson D (2017) Unsupervised early prediction of human reaching for human–robot collaboration in shared workspaces. *Auton Robot* 1–18
13. Nikolakis N, Kousi N, Michalos G, Makris S (2018) Dynamic scheduling of shared human-robot manufacturing operations. *Procedia CIRP* 72:9–14
14. Rahman SMM, Sadrifaridpour B, Wang Y (2015) Trust-based optimal subtask allocation and model predictive control for human-robot collaborative assembly in manufacturing. In: ASME 2015 dynamic systems and control conference. American Society of Mechanical Engineers, pp V002T32A004–V002T32A004
15. Robla-Gómez S, Becerra VM, Llata JR, González-Sarabia E, Torre-Ferrero C, Pérez-Oria J (2017) Working together: a review on safe human-robot collaboration in industrial environments. *IEEE Access* 5:26754–26773
16. Tsarouchi P, Matthaiakis A-S, Makris S, Chryssolouris G (2017) On a human-robot collaboration in an assembly cell. *Int J Comput Integr Manuf* 30(6):580–589

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), 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 license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license 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.