SafeNet of Unsafe Devices
Extending the Robot Safety in Collaborative Workspaces

Federico Vicentini, Nicola Pedrocchi and Lorenzo Molinari Tosatti
Institute of Industrial Technologies and Automation (ITIA), National Research Council (CNR), via Bassini 15 Milan, Italy

Keywords: Safety, Industrial Robots, Network Controlled System, System Engineering, Functional Safety, ISO 13489, ISO 10218.

Abstract: Collaborative workspaces represent the benchmark scenario of contemporary and future industrial robotics, where hybrid production systems and multimodal interactions among human operators and robots in cooperative tasks can foster the flexibility of robotic systems. Physical interactions together with dynamic workspace-sharing represent some reference applications in ISO 10218-2, where restrictive conditions for safety are posed at system level, eventually limiting the robot execution speed. With the aim of extending the use of industrial robots in shared environments and allowing the use of generically unsafe sensory and computational components for advanced applications, a methodology called SafeNet is presented. It considers the system as a device at large and applies the concept of functional safety (ISO 13489-1) with a set of architectural procedures and implementations. The safety aspects of structure, reliability and monitoring are addressed by a redundant system of computational nodes distributed over a network. SafeNet systems can be upgraded to candidate for safe Performance Levels.

1 INTRODUCTION

Collaborative workspaces (Fig. 1) are widely reckoned by both the industrial and the academic communities as one of the elective scenarios for the present-day industrial robotics. Safety, specifically, is one of the predominant functional aspects at both machine and system levels. Under this perspective, robots, as stand-alone machines, benefit from several technologies designed for a transparent and safe physical Human-Robot Interaction (pHRI) (De Santis et al., 2008; Alami et al., 2006). Such technologies support entirely new benchmarks for service robotics, as well as for many industrial applications. Examples include compliant actuation systems (Grebenstein et al., 2012; Bicchi et al., 2008; Zinn et al., 2004; Pratt and Williamson, 1995) and lightweight platforms (Kock et al., 2011; Albu-Schäffer et al., 2007a) that feature compliant behavior attained by mechanics and control (Albu-Schäffer et al., 2007b) and that display limited energy transfer in impacts (Haddadin et al., 2008; Haddadin et al., 2009). Together with internal or add-on sensing, e.g. tactile skins (Vogel et al., 2011), such compliant platforms represent a class of elective devices for shared environments. In such a context, safety issues are predominantly treated in terms of hazardous impacts or energy transfer, whose magnitudes and way of assessing are under discussion in ISO/TS 15066 (ISO, 2011c).

Physical HRI, however, is only a form of collaboration in shared workspaces. Paradigmatic workflows may, in fact, involve a mix of hand-guided procedures and contactless co-presence in the same safeguarded space. Such scenarios are particularly relevant for industrial robots, which as stand-alone devices have to comply with eventual stops or speed limitations.

Figure 1: Paradigmatic scenario of a collaborative workspace for an industrial robot cell without fences.
limitations in such safeguarded spaces, as required by ISO 10218-1 (ISO, 2011a). Many optional safety packages in commercial controllers (KUKA Robotik, 2012; ABB Robotics, 2008) are, in fact, available for joints position safe checks at runtime, providing the basic information for a safe assessment of the robot configuration within a safeguarded space. This, in turn, represents the necessary condition for integrating safe application modes (Fig. 2) in dynamically shared environments as in ISO 10218-2 (ISO, 2011b).

Nevertheless, robots and robot systems, compulsorily featuring speed limitations in the safeguarded workspace (normative status quo), may conversely need higher task speeds and, additionally, may require the use of pervasive sensing and context awareness. This monitoring capability almost always needs distributed sensor equipments dedicated to the detection of the environment and users. Sensor processing and interpretation could, in turn, require significant computational power, so that collaborative workspaces would be, in a general sense, portrayed as distributed robotic systems¹ (Fig. 3). The resulting paradigmatic architecture is therefore a network of general-purpose devices, notably including unsafe nodes and where safe/unsafe controllers are parts of a wider set of data producers/consumers.

In this paper we discuss a methodology developed in fact to fulfill ISO 10218-2 safety requirements for a robotic system with unsafe nodes (robots included) through a set of architectural and procedural actions over the system. The two key concepts are that (i) the system at hand can be seen as a single (complex) device that (ii) has to display functional safety as a whole. Functional safety is the "part of safety relating to the Equipment Under Control (EUC) and the EUC control system that depends on the correct functioning of the Electric/Electronic/Programmable Electronic (E/E/PE) safety-related systems, other technology safety-related systems and external risk reduction facilities" (IEC, 2010). Since the system at hand can be considered a single EUC when used for interacting with and monitoring the collaborative workspace, it is required to be validated with respect to functional safety criteria as in ISO 13489-1 (ISO, 2006). Equivalently, components in a system are not required to be safe per se but, rather, the system functional safety depends on to which extent the residual probabilities of failures in exchanged data can be limited.

The core methodology here discussed aims at extending the functional safety of data flows before any usage of such data in the network. Applications eventually using such safe data in safety functions do not contribute to the preliminary safe assessment of data. Rather, being the way the nodes are safely checked relevant for the overall risk assessment, such network can freely integrate both safe and un-safe sensors/devices. This would make the exclusive use of individually safety-rated devices non-necessary for a safe system integration. A relative freedom in the integration of subsystems, remarkably computational nodes in PC-based robotic applications, is considered to be beneficial for the evolution of industrial robotic cells towards fully-collaborative fully-open environments. Such freedom of components choice, sometimes actually being the only choice because of required specific technologies that are not supported by safety-rated devices, reflects the concept of extension

¹Specifically Network Controlled Systems (Gupta and Chow, 2010; Hespanha et al., 2007), when control actions proper are distributed among several nodes.
of safety features in networks rather than restricting
the usage of few rated protocols, as reckoned in the
progressive introduction/standardization of safe
protocols into the main families of automation fieldbuses
(Moyne and Tilbury, 2007; Decotignie, 2005; Felser,
2005).

On top of the methodology here discussed, the
system integration has nevertheless to provide a gen-
eral assessment, evaluation and mitigation of risks ac-
cording to ISO 12100 (ISO, 2010) wrapping guidelines,
which are out of scope in this work. In the next
section, instead, the procedural and architectural as-
pects of the extension of safety to data flows are dis-
cussed.

2 SAFENET FRAMEWORK

Functional safety is a key element of system design
based on (i) well-tried components and methods and
on (ii) the application of the principles of redundancy,
diversity, monitoring. Functional safety is expressed
as a ISO 13489-1:2006 Performance Level (PL), or
equivalent IEC 61508 SIL level, which encapsulate
the rate of reliability, failure-detectability and ready-
ness to recovery of a component/system. Specifically
for a robotic system, the required safety-rate is (ISO,
2011b):

\[
PL_r = d \quad \text{i.e. } PFH_d \in [10^{-7}, 10^0]
\quad \text{Cat. 3 Designated Architecture}
\]

where \(PL_r\) is the required performance level, \(PFH_d\)
is the Probability of dangerous Failures per Hour and

Cat. 3 is one of the two safest ISO 13489-1:2006 cat-
ergories of Safety Related Parts of a Control System
(SRP/CS) using double channels (see details in sec-
tion 2.2). Such functional safety rate is the aim of
the actions (Fig. 4-a) that transform a network of un-
safe devices in a SafeNet and that mainly involve a
double set of data validation and cross-checking. The
purpose is to reduce the probability of failing in de-
tecting occasional inconsistencies in data processed
by different nodes. Distributed systems, in particu-
lar, are likely to include sensors used for environmen-
tal monitoring that are eventually available for track-
ing the robot(s) motion inside a shared workspace as
well. The possible configurations of sensors and con-
trollers are very diverse, only optionally including na-
tive safety packages in robot control. External motion
tracking information are, instead, rarely matching the
safety-rate standards.

The above listed principles of redundancy, diver-
sity and monitoring are therefore applied to the ver-
fication of such tracking information by a double
independent elaboration of a single target informa-
tion, obtained along both a procedural and an archi-
tectural dimension (Fig. 4-b). The procedural redu-
dancy corresponds to the plain use of data from both
the tracked (unsafe) robot and the tracking unsafe sen-
tors, verifying that values match within given toler-
ances. The architectural redundancy, complementar-
ily, is obtained distributing robots and sensors data in
doubled flows for independent procedural evaluation.
Then, both comparative units (check nodes) are veri-
fied for consistency by a final safe unit/layer, i.e. data
are fed to safety functions coded according to IEC
61131-3/61508 in natively safety-rated logic devices or mapped in safe I/Os distributed over safe protocols. Such final step is compactly represented in Fig. 4-b by a safe node that acts as the safety gate between the safety functions domain and all the general purpose CPUs or unsafe sensors.

As a result, such architecture is equivalent to a SRP/CS distributed in two components and a safe node, suitable to fulfill the preliminary conditions for a PLd implementation, i.e. the dual structure and the availability of monitoring coverage.

Proceeding with the tracking configuration, the procedural and architectural aspects of the SRP/CS are discussed in the following subsections: robots and sensors data-check (procedure, Section 2.1) mainly involve kinematic and accuracy considerations, while the data-flows dispatching (architecture, Section 2.2) are considered according to ISO 13849-1:2006 guidelines.

### 2.1 Procedural Data Check

Considering a basic configuration with a robot moving along a joint trajectory \([q, \dot{q}]\), with speed \(v\) and tracked by a set of sensors (Fig. 5), each unit verifies that motion data from robots and sensors correspond, i.e. whose difference remain in a given safe interval. The motion data difference \(d_{SE3}\) can be evaluated in each check node (Fig. 6) according to any Lie-algebra consistent metrics\(^2\). Intervals and/or allowed regions for motion data verifications depend on the system and the application, e.g. largely on speed \(v\) and on the networking that may affect the data exchange. Measurement inaccuracies depend, in fact, on several factors, either spatial or temporal:

- **errors in calibration** that usually depend on the position inside the workspace due to the anisotropy of the calibration procedures

\[
\varepsilon_{\text{calib}} = \varepsilon_{\text{calib}(q)} = \varepsilon_{\text{calib}(sens)} + \varepsilon_{\text{calib}(frames)}
\]

where \(\varepsilon_{\text{calib}(sens)}\) is the intrinsic precision of the sensor and \(\varepsilon_{\text{calib}(frames)}\) is the accuracy of the hand-eye calibration (Tsai, 1987). In case of calibration procedures based on same sensors used during the tracking, the inaccuracy propagates from the sensor precision to the errors in reference frame alignments:

- **tracking errors** of the manipulators,

\[
\varepsilon_{\text{dyn}} = \varepsilon_{\text{dyn}}(q, \dot{q}, f) \simeq \varepsilon_{\text{dyn}}(q, f) \ll \varepsilon_{\text{calib}}
\]

is usually negligible in presence of accurate modeling of the residual flexibilities at both link and joint levels and proper compensatory control strategies;

- **temporal misalignment** (\(\tau\) in Fig. 5) between the sampled poses and the real pose

\[
\varepsilon_{\text{lat}} = \varepsilon_{\text{lat}}(\Delta T_{\text{sens}})
\]

where (see Fig. 7)

\[
\Delta T_{\text{sens}} = T_{\text{off-set}} + \Delta T_{\text{proc}} + \Delta T_{\text{lat}} + \sum k jT_{\text{lat}} \geq 0
\]

is the cumulative time delay due to channels asynchronicity (\(T_{\text{off-set}}\)), sensor information processing time (\(T_{\text{proc}}\)), protocol-dependent transfer latency (\(T_{\text{lat}}\)) and related jitters, that ends up into a blind time-of-flight for the robot.

The overall inaccuracy \(\varepsilon = \sum \varepsilon_k \geq \varepsilon_{\text{min}} > 0, \forall k\) sources listed above, introduces a non-null risk of erroneous tracking of the robot (risk factor, \(RF \geq RF_0\)) that increases more than linearly with \(\varepsilon\) (Fig. 8).
While spatial and control inaccuracies may be considered negligible in most of practical cases, with $\varepsilon_{calib} + \varepsilon_{dyn} \leq \frac{5}{1000}$ mm, the latency-dominated inaccuracy $\varepsilon_{lat}$ plays a significant role in building the overall blind time-of-flight $\tau$ along which the robot moves without any chance of detection (Fig. 5). The latency component, in fact, assumes the dominant role in evaluating data from sensors. Considering in fact a group $S$ of slower devices w.r.t. the group $F$ of faster devices - e.g. robots - with sampling frequency $[1-5]ms \geq T_{s}^{F} < T_{s}^{S} \in [5-20]ms$, and timing reasonably being

$$T_{nff} \leq T_{proc}^{F}$$

$$T_{s}^{S} \simeq T_{s}^{F} \in [1-5]ms$$

$$T_{s}^{S} = T_{s}^{F} + T_{comput} \gg T_{proc}^{F},$$

the sensor processing happens to be the prominent contributor to the overall time misalignment in data checking, i.e.

$$\tau \simeq T_{s}^{pro}$$

$$\varepsilon \simeq \varepsilon_{lat}.$$ 

As a result, demanding applications, e.g. fast robot motion - which is currently not allowed in standards (ISO, 2011b) - and time-expensive environmental monitoring, happen to require larger tolerances or larger uncertainty regions (e.g. larger risk factor Fig. 8) where each check node enters a safe state.

From a SafeNet procedural stand point, the monitoring principle would benefit from a reduction of such $RF$ or, correspondingly, an improvement in quality of the sensor channels. The monitoring of channels, and their quality at large, tend to limit the number of failures (i.e. $d_{SE3} > \varepsilon$) per time unit, significantly contributing to the improvement of the system reliability, which in turn is one of the steering parameters in ISO 13489-1:2006 PL assessment.

2.2 Architectural Designation and Performance Level

The set of architectural actions (Fig. 4), aimed at differentiating and doubling the data flow evaluation, provide the necessary structure of a PL $d$ class of functional safety. Architecturally this is equivalent to distributing a SRP/CS over 3 components, being able to cross-monitor the double data channels. In a ISO 13489-1:2006 Cat. 3 architecture with dual channel I-L-O (input-logic-output) modules (Fig. 9), all monitoring functions are, in fact, performed by the safety functions in the safe node.

From an implementation point of view, this can be achieved by embodying the check nodes and the safe node in 3 separate PLCs (Fig. 10) networked through any suitable protocol (chiefly Ethernet-based) to the system and mutually through a safe protocol. On
Figure 10: UML deployment diagram of a system made of a robot and a sensor set connected to the SRP/CS made of 2 standard PLCs (check node 1 and 2) and one safePLC (safe CPU). “double to double” connections from/to the SRP/CS are visible for all data feeds in each data source (data1, data 2, data ...). Watchdogs are present for DC purpose and check consistency component in SafeCPU node is in charge of handling the safe state. do algorithms component in SafeCPU node represents the data usage in a functionally safe mode, i.e. through safety functions.

data transfer protocol (e.g. port access) and (ii) the data sampling/processing routines. In particular, the L modules in both channels are directly connected to the safePLC (Fig. 11) through safe protocols, ensuring a supervised output for each channel (n in Fig. 9). The same apply at inter-logic level (cm in Fig. 9).

Finally, ISO 13489-1:2006 requirements for functional safety include also the use of (application-dependent) well-tried procedures, components and methods in system development in form of a review of measures for avoiding the common causes of failures (CCF) that have to gain a minimum score of 65 according to quantification in Tab. 1.

Table 1: Measures against CCF (common causes of failures) scores.

<table>
<thead>
<tr>
<th>measure</th>
<th>max score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separation between the safety circuits</td>
<td>15</td>
</tr>
<tr>
<td>Diversity in design and technologies</td>
<td>20</td>
</tr>
<tr>
<td>Draft / Application / Experience in applying well-tried procedures</td>
<td>20</td>
</tr>
<tr>
<td>Assessment / Analysis of developers</td>
<td>5</td>
</tr>
<tr>
<td>Environmental influences</td>
<td>35</td>
</tr>
<tr>
<td>EMC and others</td>
<td></td>
</tr>
</tbody>
</table>
3 CONCLUSIONS

A methodology has been outlined discussing procedural and architectural actions aiming at qualifying a robotic system with a functional safety rate equal (at least) to $PL_d$, as requested by ISO 10218-2:2011, in the case of entire/partial presence of unsafe nodes (Fig. 12).

The core concept introduced in such a methodology (SafeNet of unsafe devices) considers the system as a device at large, which has to display functional safety in its parts and nodes. Required level of functional safety has been reviewed to be formulated on the basis of system-level reliability and monitoring ($MTTF_d$ and $DC_{avg}$), to require well-tried and consistent practices (CCF counter-measures), and, most importantly, to stand on a class of dual channel monitored architectures where the SRP/CS is able to consistently check the availability and validity of data feed and component behaviors. Such structural feature is obtained through the main characteristic of the SafeNet that involves the creation of procedural and architectural redundancies over the network, variously interconnecting robots and sensors. In this way, general systems of designated architectures Cat.$B/1/2$ can be upgraded to Cat.3 and can provide necessary conditions for $PL_d$ rate achievement (Fig. 13). The safety rate upgrade is mainly in charge of a SRP/CS distributed on 3 components that provide the structure for the designated category as well as the reliability and diagnostic coverage.

ACKNOWLEDGEMENTS

This work has been partially supported by CNR Flagship Program “Factory of the Future”, FdF-SP1-T3.1, Project FACTOry Technologies for HUMans Safety.