ROBOT SAFETY
Proceedings of the Symposium on Robot Safety
April 7, 1989, TU Delft
Robot Safety
F. Koornneef (Ed.)

Proceedings of a Symposium on Robot Safety

Organised by the Electrical Engineering & Safety Group
Delft University of Technology
CONTENTS

Preface 5

G. Honderd, Developments in Robot Control 7

L.N. Reijers, State-of-the-Art of Robot Manufacturing Systems 9

L.H.J. Goossens, Risk Assessment of Robot Systems 19

P.L. Latour, Automatic Robot System for Container Transportation 39

J.J. Gebrants, E. Backer and P.P. Jonker, Vision in Robots 49

A. van Buuren, Considerations of applying Communication Standards for Robot Control 61

Electrical Engineering & Safety Science Group 71
Preface

After the exploring Symposium on Functional Safety of Programmable Electronic Systems in 1987, the Electrical Engineering & Safety Group at the Delft University of Technology has moved into the field of safety-critical or availability-critical (real-time) computer controlled systems.

The initiative to organise a symposium on robot safety was born after consideration of the following arguments:
- Robot systems are, in most cases, controlled by computers. Depending on their application, they encompass risks reflecting safety and availability.
- Robotics are a focal point in the research activities of the Department of Electrical Engineering.

The purpose of would be to overview developments in robotics in order elaborate risk problems and roads to solutions for risk control of robot systems. The contributions show that so far more emphasis is given to fail-safe control strategies and less to fault-tolerance which would put higher demands on the control computer sub-systems.

This booklet contains the submitted papers for this symposium. One may notice that not all lecturers were able to submit a paper due to heavy workload. In our view, the remaining papers do still offer an adequate coverage of the symposium topics.

Support has been given to the symposium by the Royal Institution of Engineers in the Netherlands KIVI, The Dutch Association of Safety Engineering NVVK and the Society of Safety Science GVW, which attributed to valuable exchange of ideas and presentation of advanced topics as reflected in the symposium booklet.

Special thanks go towards the Programming Committee (Dr.ir. L.H.J. Goossens, Prof.ir. G. Honderd, J. van Laren and ir. F. Koornneef), to mrs. Y. Smits who took care of the symposium logistics, and to mrs. E. v. Verseveld who produced all the paperwork, for their work in organising this symposium.

Dr.ir. R.P. van Wijk van Brievingh

(chairman of the Electrical Engineering & Safety Group)
A robot can be defined as a reprogrammable multi-functional manipulator designed to move materials, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks. In the present interpretation a robot is a reprogrammable, flexible, general-purpose manipulator with external sensors which can perform various tasks. This means that a robot performance must include a certain "intelligence", normally implemented by computer algorithms, associated with its sensing systems and its control.

Recent developments in robot control are concerned in a broad area of disciplines like artificial intelligence, smart sensors, vision as well as non-vision, adaptive & optimal control and kinematic transformations.

The first generations of robots were stiff, inflexible and difficult to reprogram. Safety measures were generally taken by introducing a safe-guarded working space.

More and more, robots are introduced as an extension of human beings. The robot actions are easily programmable in terms of goal-oriented tasks: the specific trajectory is calculated by the robot computer, based on the criteria for a desired performance, as input by the operator or user. Certain dangerous situations for the robot as well as for its environment are detected and adequate control actions can be taken. This development stresses the necessity to include safety measures in the performance criteria of a robot.

For industrial robots as well as AGV's a new paradigm, called robotics, has been established, including computer science, control, electronic instrumentation and mechanics. The introduction of human-oriented knowledge for the performance of certain class of tasks under a more or less reliable external sensor feedback gives rise to specific research projects, which will be discussed in this survey.
Industrial robots are production machines which are used in increasing numbers in companies that manufacture discrete products. Robots are economically most attractive in series production with annual series of less than one million products, manufactured in a large variety. This is where their flexibility is needed. Even though the industrial robot concept is universal in nature, robots appear in a large number of configurations, which are more or less process-specialized. In the large family of manipulators for industrial use, only the industrial robot is truly fully programmable.

On one hand industrial robots increasingly replace the classical fixed mechanisation in mass- and large series production. On the other hand robots replace human labour: simple manipulative tasks that are repetitive in nature. However at the present only a fraction of the human tasks that essentially can be automated, are performed by industrial robots. The reason is that the present generation of industrial robots are insufficiently equipped with sensor systems and the accompanying intelligence, to act upon unforeseen conditions. Too much human supervision is still necessary. Also industrial robots are difficult—and therefore costly—to program. Finally the cost of engineering of an automated workstation is still very high.

These are the reasons why at the present most robot applications are economically marginal.

In the world, most robots can be found in automotive industry, where spotwelding is their primary task.

Other important industrial applications are:
- electric arc welding
- product handling
- coating
- deburring and polishing.

A minority of industrial robots are used in assembly tasks. However this is a rapidly growing application area. In most manufacturing plants, assembly is where most manual labour is spent. The potential for industrial robots here is very large, but a number of serious problems must be solved first, such as:
most present day products are not designed for automated assembly
- assembly peripherals - such as part feeding equipment - that are truly flexible are not yet available in the market
- designing and programming flexible assembly cells is tedious and costly
- assembly cells need too much supervision, because they lack "intelligence" to deal with deviations in the production process.

One year ago, at the Delft University of Technology a large interdisciplinary research project in the area of automated assembly was set up. It is called the "Delft Intelligent Assembly Cell" (DIAC). Total expenditures f 10 mln. in four years, 37% of which is funded by a government computer science project (SPIN). It unites 8 research groups from 4 faculties (Mechanical Engineering, Applied Physics, Electrical Engineering and Computer Science), which have committed a total of 75 man-years. The common goal of this complex project is simple: within four years build an intelligent assembly cell which is capable of assembly at least two industrial products.

Finally I want to make a remark on the subject of this conference: robot safety. Robots fit in the class of programmable production machines. Contrary to what is often stated, they do not pose a basically different or new safety problem, compared to other machines, such as numerically controlled machine tools. New elements may be that the work area of many robots is large compared to their size and is less confined than that of NC machine tools. Also the teach-in programming method, peculiar to robots, requires the programmer to be in the robot work-area.

REFERENCES
Figure 1: types of manufacturing

<table>
<thead>
<tr>
<th></th>
<th>Added Value (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass production</td>
<td>7.5</td>
</tr>
<tr>
<td>Large series production</td>
<td>32.5</td>
</tr>
<tr>
<td>Small series production</td>
<td>60</td>
</tr>
</tbody>
</table>

Figure 2: types of manufacturing: distribution

Figure 3: the family of manipulators
Figure 4: position and orientation of the wrist

Figure 5: flexible automation
State-of-the-Art of Robot Manufacturing Systems

- Cartesian gantry type
- Cartesian arm type
- Swivel arm type
- SCARA gimbal type
- Articulated arm type

Figure 6: Robot configurations

- Production resource
- Example

- Humans
  - Manual assembly

- Universal machine
  - Stand-alone industrial robot

- Process specialised machine
  - Robot welding cell
  - NC-machine tool

- Product specialised machine
  - Mechanisation

Figure 7: Flexibility versus productivity
State-of-the-Art of Robot Manufacturing
Systems

State-of-the-Art of Robot Manufacturing Systems

- Total no. of installed robots and robot density*

<table>
<thead>
<tr>
<th>Country</th>
<th>1985</th>
<th>1986</th>
<th>1986</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRD</td>
<td>8800</td>
<td>12400</td>
<td>14.4</td>
</tr>
<tr>
<td>France</td>
<td>3950</td>
<td>5273</td>
<td>14.2</td>
</tr>
<tr>
<td>Italy</td>
<td>4640</td>
<td>5000</td>
<td>10.2</td>
</tr>
<tr>
<td>Sweden</td>
<td>2046</td>
<td>2363</td>
<td>38.7</td>
</tr>
<tr>
<td>U.K.</td>
<td>3208</td>
<td>3683</td>
<td>5.9</td>
</tr>
<tr>
<td>Belgium</td>
<td>875</td>
<td>1050</td>
<td>9.6</td>
</tr>
<tr>
<td>Spain</td>
<td>675</td>
<td>854</td>
<td>2.3</td>
</tr>
<tr>
<td>Netherlands</td>
<td>389</td>
<td>508</td>
<td>6.2</td>
</tr>
<tr>
<td>Finland</td>
<td>251</td>
<td>336</td>
<td>4.3</td>
</tr>
<tr>
<td>USA</td>
<td>20000</td>
<td>27000</td>
<td>8.2</td>
</tr>
<tr>
<td>Japan</td>
<td>41000</td>
<td>90000</td>
<td>?</td>
</tr>
</tbody>
</table>

* no. of industrial robots per 10,000 employees in industry

Figure 8: installed base by country

Robot applications in the Netherlands in 1986

- Arcwelding 29%
- Loading - unloading
- Handling, palletising 23%
- Spotwelding 17%
- Coating, bonding 9%
- Assembly 9%
- Education 9%
- Other applications 4%

Source: TU Delft

Figure 9: application of industrial robots
Figure 10: sequential robot configuration
Figure 11: assembly cell layout
Figure 12: data flows in the assembly cell
Risk Assessment of Robot Systems

L.H.J. Goossens
Delft University of Technology
Safety Science Group
Delft, The Netherlands

Abstract

Past experience with scarce detailed data on robot accidents reveal eight typical accident scenarios, which enclose in principle all robot accidents in practical situations. The characteristics of such accident scenarios with respect to accident prevention strategies will be discussed.

As a practical application the safety implementation of a large robot park in a car manufacturing system will be used to demonstrate a two-step risk assessment procedure. The first step is a deterministic approach implementing several safety policies. The second step is a probabilistic approach which deals with the unacceptable residual risks of the deterministic approach.

1 INTRODUCTION

How do we perceive robot safety? Fatal accidents with such a relatively new production system are in the forefront of people's attention. This is particularly so in cases of resistance against such a new technique. How many accidents really occur? And even more interesting, how many accidents are there going to occur in the near future as a result of rapid growth of the robot park? Or can we diminish the accident rate? Even close to zero?

The paper presents a risk assessment procedure which may help the robot user in order to prevent accidents or at least give them a clue how to treat the problem in a time-effective way. The paper consists broadly of two parts. The first part analyses potential accidents by defining eight accident scenarios which effectively cover all possible mishaps within robot use. The second part analyses residual risks for each accident scenario and formulates prevention strategies, both in a deterministic way and in a probabilistic way.

2 WHAT CAN WE LEARN FROM PAST EXPERIENCE?

What accidents are reported? From the Swedish ISA-databank (accident reporting system) recently about 7 to 8 accidents with days-off are reported on an annual
Risk Assessment of Robot Systems

L.H.J. Goossens

basis. The Japanese have reported in 1987 over 10 fatal accidents in total. In the Netherlands 3 smaller accidents have been reported officially up to 1987. What do these figures mean? In order to compare one should at least know how many robots were effectively in operation during the years considered. Table 1 can be reproduced from somewhat older Japanese data (JISHA, 1983), in which it is assumed that on average every workplace has the same amount of robots (because they did not report the growth in robots, only in workplaces).

<table>
<thead>
<tr>
<th>year</th>
<th>#workplaces</th>
<th># robots estimated</th>
<th># accidents</th>
<th># acc./robot/yr</th>
<th># near-accidents</th>
<th># near-acc/robot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1978</td>
<td>59</td>
<td>1350</td>
<td>2</td>
<td>0.0015</td>
<td>2</td>
<td>0.0015</td>
</tr>
<tr>
<td>1979</td>
<td>75</td>
<td>1720</td>
<td>2</td>
<td>0.0012</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1980</td>
<td>105</td>
<td>2400</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>0.0008</td>
</tr>
<tr>
<td>1981</td>
<td>158</td>
<td>3520</td>
<td>6</td>
<td>0.0017</td>
<td>13</td>
<td>0.0037</td>
</tr>
<tr>
<td>1982</td>
<td>190</td>
<td>4341</td>
<td>1</td>
<td>0.0005</td>
<td>20</td>
<td>0.0092</td>
</tr>
</tbody>
</table>

Table 1: Overview of robots and (near-)accidents in Japan (adopted from JISHA, 1983)

The 1982 data only cover the period January - June.

Table 1 shows an accident rate of about 1 in a 1000 robot years and a near accident rate which is rapidly growing. What the table does not say, unfortunately, is what the effect of preventive measures is.

Let us take another figure. From the above mentioned data the number of fatal accidents was reported to be 2 in 1982, having 4341 robots in operation. In 1987 10 fatal accidents have been reported in total with a robot population of 80,000 to 100,000. If the fatal accident rate would be constant one would have expected 40 fatal accidents to be reported in 1987. However, there were only 10. Assuming both a constant growth rate of robots, and a constant fatal accident rate this leads to a growth rate of robots of 0.6 per year and a fatal accident rate of 0.33 per year. But are these rates constant? One needs more data of the years in-between and of course of the years to come.

So, effectively, all the above mentioned figures really do not say much about robot safety. One sort of starting point can be drawn from it, which is that an accident frequency of 0.001 per robot per year is a reasonable figure to use as a reference value throughout the paper.

Robot accidents, as all accidents do, follow McDonald's scheme (for further details, see Hale and Glendon, 1987), somewhat revised shown in figure 1. Damage to people occurs due to accidents. Accidents occur due to fail-to-danger.
event, because a potential unsafe situation could not be prevented by fail-to-safe measures. Such an unsafe situation occurs because deviations of the operational system could not be controlled within the operational system. The concept of deviations has been described by Kjellén and Larson (1981) and restated in a risk analysis format by Suokas (1988).

Deviations are defined as events or conditions in the production process conflicting with the norm for the faultless and planned process. So a deviation is a change in the operation of the production process, but it does not necessarily lead to the occurrence of an accident. Deviations expose themselves as events within the time frame of the production system under consideration. Deviations have a duration of fractions of seconds to several weeks provided the time frame considered is much larger than that.

There are, in general, background factors which serve as necessary constituents for accidents too. These are the determining factors. These factors are of
physical/technical, organisational/economical and social/individual nature and have a larger time span than the time frame considered. The determining factors are built into the process (system concept, design, installation, commissioning, operating procedures). The frequency of occurrence of deviations depends greatly on the determining factors.

Looking at accident descriptions one can see that as many as 10 to 20 deviations can be present with as many determining factors. Obviously, in-depth accident analyses give a good view on the system from a safety point of view, but for risk assessment purposes it is very time-consuming and does not reflect future developments with the robot system sufficiently. It, however, provides past experience, which must be used within a risk assessment!

3 ACCIDENT SCENARIOS AND RISK

What we learned up till now is that the overall statistics of robot accidents only provide us with a reference value of the accident frequency, and the accident investigations are either too numerous or too few, and too specific. What one needs for risk assessment purposes is something in-between: clustering of accident descriptions and converging to not too many. This leads us to the concept of accident scenarios. From all information collected (accident descriptions and workplace observations, mostly provided by the Swedish ISA-databank: see Bonney and Yong, 1985) we arrived at the following accident scenarios presented in a verbatim way in table 2.

If we were to consider these accident scenarios as generic accident descriptions one can see that only a small part of the possible deviations and determining factors can be deduced from them. E.g. the only deviation discovered in $s_1$ is a deviation of the human operation in time and space, while the only determining factor is a normally operating robot containing an energy flow which could cause the accident, because a human being approaches the robot. As we shall see later, this description is enough for prevention strategies in a deterministic risk assessment. It, however, does not state anything further on the underlying causes of possible accidents.
The risk assessment procedure should be more supported by identifying the relevant risk factors. If one knows the potential contribution of such a risk factor to the accident frequency one is able to distinguish the relevant prevention measure. This is where the accident scenario concept intervenes the prevention strategy concept.

**Risk factors** are defined using a system model of the production systems in which robots are installed. The system model consists of five basic risk factors to be subdivided (table 3). For further details of the robot accident scenarios, see Goossens and Hoefnagels (1989). Further elaboration of the accident scenario concept can be found in Heimplaetzer et al (1988).

---

<table>
<thead>
<tr>
<th>accident scenario</th>
<th>verbatim description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>The operator comes into contact with the robot arm, being within reach of the normally operating robot.</td>
</tr>
<tr>
<td>$S_2$</td>
<td>The operator comes into contact with the robot arm or any work piece or tool mounted in the gripper, during changing work pieces or tools at a normally operating robot.</td>
</tr>
<tr>
<td>$S_3$</td>
<td>The operator comes within reach of the robot and comes in contact with the robot arm due to an unexpected motion of the robot arm.</td>
</tr>
<tr>
<td>$S_4$</td>
<td>The programmer comes into contact with the robot arm during programming within reach of the robot.</td>
</tr>
<tr>
<td>$S_5$</td>
<td>The maintenance man comes into contact with the robot arm or gets caught between moving parts during repair.</td>
</tr>
<tr>
<td>$S_6$</td>
<td>The operator, maintenance man, programmer or by-stander gets hit by an object projected by the robot.</td>
</tr>
<tr>
<td>$S_7$</td>
<td>The operator, maintenance man, programmer or by-stander gets touched by a material or energy stream originating from the production process.</td>
</tr>
<tr>
<td>$S_8$</td>
<td>A third party comes within reach of the robot and comes into contact with a moving robot arm.</td>
</tr>
<tr>
<td>$S_9$</td>
<td>Any other unforeseen robot accident, including non-robot related accidents at the robot system (like falling).</td>
</tr>
</tbody>
</table>

Table 2: accident scenarios for robot accidents
The risk of occurrence of accidents is defined by Kaplan and Garrick (1981) as a set of triplets

\[ R = < s, p, x > \]

where \( R \) stands for risk, \( s \) for accident scenario, \( p \) for probability of occurrence of that scenario and \( x \) stands for the consequences in terms of damages. Formula (1) describes the risk factors mentioned under RE of table 3. Accident descriptions generally add RD-type risk factors to the analysis of \( R \). It means that one not only knows data, but also what sort of interaction with the robot system took place in the final phase of the accident.

Obviously, it is very interesting to know whether certain "higher-order"-risk
factors, in particular the ones of type RB, contribute or dominate the accident process. In cases of preventive measures one should then not try to prevent interaction between human beings and the robots, but one should try and redesign the system such that interactions are eliminated.

This can be done by investigating risk factors (called briefly Rf) in the light of the occurred accidents. In other words by investigating the formula

\[
p( R_f | \text{acc} \cap R_f )/p(R_f | \text{acc})
\]

- 1 means no significant influence
- > 1 means Rfx has accidental potential
- < 1 means Rfx has preventive potential

within the context of accidents, Rfy being present. It provides a possibility of weighing risk factors relative to each other.

Let us now take two examples how this may contribute to risk assessment. We use Carlsson's 36 robot accidents (Carlsson, 1985).

Let us say, we are interested to know the potential of the various types of personnel working with robots, of having an accident with respect to different industries where robots are applied. We then consider the risk factor RB5 in the context of accidents occurring in companies RB1 (see table 3). We rewrite formula (2) as

\[
p( RB_1 | \text{acc} \cap RB_5 )/p(RB_1 | \text{acc}) > 1
\]

Table 4 shows that special care should be paid to maintenance personnel in the car manufacturing industries, to programmers in the metal industries and to operators in the other industries.

Of course, all depends on the knowledge about accidents in relation to the various risk factors and the number of accidents. In this respect Carlsson's data are actually insufficient. But they serve as an illustration. What is lacking are, of course, data of frequencies of use of robots (risk factors RC of table 3).
Another example may be the interest we have in knowing the influence of various types of disturbances in the process causing an effective approach of personnel to a robot. Again we rewrite formula (2) into

\[(4) \quad p( RB_1 \mid \text{acc} \cap RD_{123} ) / p( RB_1 \cap \text{acc} ) > 1 \]

in which the risk factors \( RD_{123} \) are respectively:

\( RD_1 \): these are disturbances in the human being-robot relation, indicating that the system is sensitive to either intrusion of human beings and to disturbances in the robot subsystem

\( RD_2 \): these are disturbances in the robot-production line relation, indicating that the system is sensitive to disturbances in the production sequence

\( RD_3 \): these are disturbances in the human being-production line relation, indicating that the system is sensitive to disturbances in the production line itself.

The results are shown in table 5.

<table>
<thead>
<tr>
<th>Industries</th>
<th>personnel</th>
<th>formula (3)</th>
<th>&gt; 1 ??</th>
</tr>
</thead>
<tbody>
<tr>
<td>car manuf.</td>
<td>operators</td>
<td>8/9</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>maintenance</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>programmers</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>metal ind.</td>
<td>operators</td>
<td>1/3</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>maintenance</td>
<td>9/16</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>programmers</td>
<td>9/4</td>
<td>yes</td>
</tr>
<tr>
<td>other ind.</td>
<td>operators</td>
<td>40/33</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>maintenance</td>
<td>9/11</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>programmers</td>
<td>0</td>
<td>no</td>
</tr>
</tbody>
</table>

Table 4: accidental vs preventive potential for various types of personnel for the various industries
<table>
<thead>
<tr>
<th>industries</th>
<th>disturbances</th>
<th>formula (4)</th>
<th>$\geq 1$ ??</th>
</tr>
</thead>
<tbody>
<tr>
<td>car manuf.</td>
<td>in robot</td>
<td>$1/2$</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>in sequence</td>
<td>$3/2$</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>in production</td>
<td>$2/3$</td>
<td>no</td>
</tr>
<tr>
<td>metal ind</td>
<td>in robot</td>
<td>$81/64$</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>in sequence</td>
<td>$0$</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>in production</td>
<td>$21/16$</td>
<td>yes</td>
</tr>
<tr>
<td>other ind</td>
<td>in robot</td>
<td>$27/44$</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>in sequence</td>
<td>$45/22$</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>in production</td>
<td>$9/11$</td>
<td>no</td>
</tr>
</tbody>
</table>

Table 5: accidental vs preventive potential for various types of personnel in cases of all kinds of disturbances in various industries

4 PRACTICAL IMPLEMENTATIONS OF SAFETY POLICIES

Let us now take the safety policy of a large manufacturing industry (Volvo Car B.V. Born) presented previously at this symposium (Coenen, 1989). First we compare the safety policies by a deterministic approach of accident prevention. The philosophy behind the Volvo approach has been well worked out, and the following description may look as though it violates this philosophy. This is not the intention of this paper, but by simplifying the policy a bit, it shows better how the deterministic approach works.

One basic line of thinking of Volvo is that robots may not be approached by human beings who have no reason to approach them. This effectively led to enclosures of certain dimensions. Let us take this policy very precisely. It means that all robots must be enclosed fully without any unauthorised entrance possibilities. We call this policy POL1. The effect on the accident scenarios is shown in table 6.
Let us now review how this policy POL1 effectively influences safety. Enclosures are barriers. In figure 1 barriers are shown to be part of the operational robot system. If entrances are left open unauthorised there may be only a minor decrease in the probability of occurrence of $s_1$. Effectively this requires a safety policy with respect to locking entrances (called POL2).

<table>
<thead>
<tr>
<th>accident scenario</th>
<th>consequences of POL1</th>
<th>estimated decrease of accident probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>entrance of operators is restricted</td>
<td>* no unauthorised entrance $p \to 0$</td>
</tr>
<tr>
<td>$s_2$</td>
<td>change of tools outside enclosure</td>
<td>* needs safety precautions</td>
</tr>
<tr>
<td>$s_3$</td>
<td>entrance of operators is restricted</td>
<td>* authorised entrance (for small disturbances) needs safety precautions * monitoring of robot sequence needs safety precautions</td>
</tr>
<tr>
<td>$s_4$</td>
<td>entrance of programmer is restricted</td>
<td>* programming still needs safety precautions</td>
</tr>
<tr>
<td>$s_5$</td>
<td>entrance of maintenance worker is restricted</td>
<td>* maintenance work still needs safety precautions * enclosure may act as unsafe barrier in cases of emergency</td>
</tr>
<tr>
<td>$s_6$</td>
<td>projection of objects to be kept within enclosure</td>
<td>* requires additional safety precautions on enclosures * monitoring of robot sequence needs safety precautions</td>
</tr>
<tr>
<td>$s_7$</td>
<td>material/energy streams to be kept within enclosure</td>
<td>* requires additional safety precautions on enclosures * monitoring of robot sequence needs safety precautions</td>
</tr>
<tr>
<td>$s_8$</td>
<td>entrance of third party is forbidden during any robot sequence</td>
<td>* no unauthorised entrance $p \to 0$</td>
</tr>
<tr>
<td>$s_9$</td>
<td>not taken into account</td>
<td></td>
</tr>
</tbody>
</table>

Table 6: effects of POL1 on accident scenarios
This policy POL2 is also required in order to prevent the accident scenarios $s_3$, $s_4$ and $s_5$.

Furthermore enclosures require that operators or maintenance personnel and programmers should still have a good view on the production process from outside the enclosures. This policy is called POL3.

In order to prevent accident scenario $s_2$ a safety policy is defined for changing tools and so on (called POL4).

Uncontrolled motions of the robot and objects and streams should be separately dealt with in a safety policy called POL5. This is to prevent accident scenarios $s_3$, $s_6$ and $s_7$.

In order to prevent an increase in accidents the negative aspects of enclosures must be treated in a safety policy POL6. That is to prevent accident scenario $s_5$.

Programmers need a special safety policy in order to prevent accidents during programming while being within reach of the robot (POL7).

Table 7 shows which safety policies are defined for preventing what accident scenario.

<table>
<thead>
<tr>
<th>accident scenario</th>
<th>safety policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>POL1, POL2, POL3</td>
</tr>
<tr>
<td>$s_2$</td>
<td>POL1, POL4, POL5</td>
</tr>
<tr>
<td>$s_3$</td>
<td>POL1, POL2, POL5</td>
</tr>
<tr>
<td>$s_4$</td>
<td>POL1, POL2, POL3, POL7</td>
</tr>
<tr>
<td>$s_5$</td>
<td>POL1, POL2, POL3, POL6, POL7</td>
</tr>
<tr>
<td>$s_6$</td>
<td>POL1, POL5</td>
</tr>
<tr>
<td>$s_7$</td>
<td>POL1, POL5</td>
</tr>
<tr>
<td>$s_8$</td>
<td>POL1, POL3</td>
</tr>
<tr>
<td>$s_9$</td>
<td>not specified</td>
</tr>
</tbody>
</table>

Table 7: safety policies and accident scenarios

Let us now review the various policies in the light of prevention of the 8 accident scenarios:
S1. POL1 prevents operators to approach the robot provided POL2 is effective; this requires compliance of the rule of using the lock system on the enclosure (a visit in a foreign country once showed how delicately such locks can disfunction); if POL3 provides the possibility of overlooking the robot operations from a higher level (platform), the need for approaching the robot will be substantially lower;

S2. POL1 requires changing of tools outside the robot range and requires special holes in the enclosure; POL5 may be turn tables whereby the operators change tools on the robot-free zone, appears to be a well designed solution; if that is not within reach specific safety precautions must be taken, like light curtains or safety mats which put the robot into a stop and standby-mode; these, however, may fail on demand; this may also be the case with the end-switches or approach-switches of the robot; so monitoring of the robot sequence may be required as well (POL5);

S3. POL1 leads to an effective entrance procedure guarded by POL2; this means that entrance is only possible if the power supply of the robot is fully broken, which device makes restarting of the robot from outside impossible; additional monitoring of the robot sequence can be considered if there are reasons to believe that failures in the cut-off of the power supply are possible (POL5);

S4. POL1 and POL2 provide the necessary precautions to prevent programmers to enter the robot suddenly; not all programming work should be done while being within reach of the robot, so POL3 may provide programmers a bright view on the system for that purpose; in cases programming needs to be done within reach of the robot POL2 provides a special lock position which enables the robot to operate under reduced speed; POL7 requires additionally a deadman's button and an emergency button;

S5. POL1 and POL2 provide maintenance people to enter the robot suddenly and provide a means to interrupt the robot motions definitively; POL3 again gives the possibility of an outside overview; POL6 is an additional measure which allows maintenance people to work without hindrance of the robot installation, e.g. a safe distance of at least 80 centimeter between enclosure and farthest robot point; the practical philosophy is that the robot is out of operation during maintenance work and can be tested while the maintenance worker is outside the robot area again; the locking system, however, leaves room to set the robot in slow motion again, like under programming; POL7 may be an additional safety precaution against such violations of the rules;

S6. Additional physical precautions at relevant places on the enclosure should prevent material to go outside the robot area;
s₇ like s₆, but material or energy streams may be more difficult to control;

s₈ again POL1 provides third parties to enter the robot area, while POL3 offers the possibility of a view on the installation from above.

Summarizing the following can be concluded (table 8).

<table>
<thead>
<tr>
<th>accident scenario</th>
<th>prevention strategy</th>
<th>residual problems</th>
<th>residual risk</th>
<th>further action</th>
</tr>
</thead>
<tbody>
<tr>
<td>s₁</td>
<td>isolation</td>
<td>compliance of lock procedure</td>
<td>low</td>
<td>training</td>
</tr>
<tr>
<td>s₂</td>
<td>isolation</td>
<td>turn table</td>
<td>low</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>protection</td>
<td>curtains, mats switches</td>
<td>??</td>
<td>probabilistic approach</td>
</tr>
<tr>
<td>s₃</td>
<td>isolation</td>
<td>cut-off failure</td>
<td>low</td>
<td>none</td>
</tr>
<tr>
<td>s₄</td>
<td>protection</td>
<td>unexpected motion, reduced speed failure, hardware failure</td>
<td>??</td>
<td>probabilistic approach</td>
</tr>
<tr>
<td>s₅</td>
<td>isolation</td>
<td>lock procedure</td>
<td>low</td>
<td>training</td>
</tr>
<tr>
<td>s₆</td>
<td>isolation</td>
<td>full enclosure</td>
<td>low</td>
<td>enclosure maintenance</td>
</tr>
<tr>
<td>s₇</td>
<td>isolation</td>
<td>full enclosure</td>
<td>low</td>
<td>enclosure maintenance</td>
</tr>
<tr>
<td>s₈</td>
<td>isolation</td>
<td>full compliance</td>
<td>low</td>
<td>none</td>
</tr>
</tbody>
</table>

Table 8: results of deterministic approach to robot safety measures

Concluding one can say that all accident scenarios are reasonably dealt with except for scenarios s₂ and s₄. Within the context of the risk assessment procedure there may be reasons to review the residual risk of these scenarios whether it is still acceptable. This can then be done with a probabilistic approach because the residual risk very much depends on the intrinsic behaviour of the robot sequences and on the reliability of additional safety measures. The concept of additional safety measures is a usual strategy of increasing robot safety applying European standards (Meffert, 1988; DIN, 1989).
5 ADDITIONAL PROBABILISTIC RISK ASSESSMENT

First we compare the residual risk obtained through the deterministic approach with current data. In the introduction we found Japanese figures with an accident frequency of approximately 0.001 per robot per year and a near-accident frequency of about 0.01 per robot per year, albeit data from 1982 and earlier. Volvo has used the described safety procedures for about 3 years now on their 120 robots. As there have been no accidents yet, the accident frequency must lie below 0.003 per robot per year. No knowledge is available on near-accidents.

The probability of a robot accident \( p(\text{acc}) = p \) from formula (1) is defined as

\[
(5) \quad p(\text{acc}) = \sum p(s_i) = \sum p_i
\]

From section 4 it can be shown that only two accident scenarios shall dominate, so the probability of an accident can be approximated by

\[
(6) \quad p(\text{acc}) = p(s_2) + p(s_4) = p_2 + p_4
\]

And it should be so, that in the case of Volvo Car B.V.

\[
(7) \quad p_2 + p_4 < 0.003 \text{ per robot per year}
\]

The question arises as to how far are they from this figure and is the result acceptable or not. Let us concentrate on accident scenario \( s_4 \) as an example.

From figure 1 it can be seen that

\[
(8) \quad p_4 = p_4(\text{IE}) \cdot p_4(\text{ftdIE})
\]

The initiating event in this case is an energy flow either originated from an unexpected motion of the robot due to a programming error, or a full speed motion in an expected trajectory due to a hardware or software error in the robot itself, while a programmer is within reach of the robot. If we designate the unexpected motion by \( \text{UM} \) and the full speed motion by \( \text{FS} \) the probability of occurrence of the initiating event \( \text{IE} \) is then approximated by

\[
(9) \quad p_4(\text{IE}) = p(\text{UM}) + p(\text{FS})
\]

The occurrence of an accident falling under this scenario is then described by

\[
(10) \quad p_4 = p(\text{UM}).p(\text{ftdUM}) + p(\text{FS}).p(\text{ftdFS})
\]
where ftd stands for fail-to-danger.

First we examine the base rates for UM and FS. For the unexpected motion it can be shown that

\[
(11) \quad p(UM) = p(UM\mid PR).p(PR) + p(UM\mid PR').p(PR')
\]

where PR stands for every time a programming action is carried out and PR' stands for the time there is no programming action. Since the unexpected motions due to a programming error are eliminated before the robot is cleared for production, the term \(p(UM\mid PR')\) will be very low and can be set equal to zero as a first approximation. So formula (11) simplifies to

\[
(12) \quad p(UM) = P(UM\mid PR).p(PR)
\]

\(p(PR)\) represents the number of times a robot is programmed per year. \(p(UM\mid PR)\) gives the probability that during each programming action an unexpected motion is programmed erroneously which contributes effectively to the initiating event.

For the full speed motion an equivalent formula as formula (11) can be derived, which again reduces to a formula similar to formula (12), since again there is no programmer within reach when full speed is accomplished during non-programming moments. So

\[
(13) \quad p(FS) = P(FS\mid PR) . p(PR)
\]

The full speed error is related to the reliability of the robot and thus proportional to the reciprocal of the Mean-Time-Between-Failure of the robot. Since robots in car manufacturing are reasonably reliable concerning the robot itself a relatively high MTBF may be a reasonable estimate. \(p(FS\mid PR)\) actually describes the probability of a full speed error given the time of programming. The time for programming can be seen as the relevant time interval \(T_1\) within which the full speed error should not occur. After that time interval the full speed program renders the system to a sort of repaired state. This means that the probability of a full speed error during programming can be approximated by

\[
(14) \quad p(FS\mid PR) = (T_1 . \alpha) / (2 . MTBF)
\]

We then look at the fail-to-danger terms of formula (10). What sort of safety precautions and recovery actions are possible? In figure 2 the relevant event tree is shown.
In cases we consider unexpected motions (ftd\text{UM}), we can state the following:

a. the programmer probably observes UM quite often when it occurs
b. he is able to escape quite often due to the low speed
c. hardware failures of the deadman's button or emergency button are rather low
d. being in slow motion the time-to-stop of the robot arm is enough in cases the hardware functions properly.

This means that the probability of an accident is reasonably low, which means that

\begin{equation}
    p(\text{ftd} \mid \text{UM}) = \text{low}
\end{equation}

If we consider the full speed error term (ftd\text{IFS}), we can then state the following, using the same event tree of figure 2:

a. the programmer probably does not observe the full speed motion in time since the robot does follow its programmed trajectory, albeit with a too high speed
b. if he observes, he is probably not really able to escape
c. hardware failures are exactly the same as for UM
d. being at full speed the time-to-stop of the robot arm is in general not enough.
This means that the probability of an accident is reasonably high given a full speed error. In other words, the safety precautions are less than adequate. This will probably occur in more than half of the cases, which means that an estimation of

\[(16) \quad p(\text{ftdIFS}) = 1\]

is very reasonable.

Now we can compute the probability of occurrence of accident scenario \(S_4\) with formula (10) using formulas (12), (14), (15) and (16). Table 9 gives a fictitious numerical example. These are the subjective estimates of the author, not reflected yet by operational experience.

<table>
<thead>
<tr>
<th>term</th>
<th>description</th>
<th>subjective estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>(p(\text{UM</td>
<td>PR}))</td>
<td>formula (12)</td>
</tr>
<tr>
<td>(p(\text{FS</td>
<td>PR}))</td>
<td>formula (14)</td>
</tr>
<tr>
<td></td>
<td>(a = 0.004)</td>
<td>(T_i = 1) hour</td>
</tr>
<tr>
<td>(p(\text{PR}))</td>
<td>formula (12)</td>
<td>10 per year per robot</td>
</tr>
<tr>
<td>(p(\text{ftd</td>
<td>UM}))</td>
<td>formula (15)</td>
</tr>
<tr>
<td>(p(\text{ftd</td>
<td>FS}))</td>
<td>formula (16)</td>
</tr>
</tbody>
</table>

* Comparable to 10 percent (estimated to be the percentage during programming) of the percentage of programming mistakes reported by Sugimoto (1985) and the percentage of unexpected motions reported by Jones and Dawson (1985).

** Sugimoto (1985) reports a MTBF > 1000 hours for 25 percent of robot population; Vermeulen (1988) reports a percentage of only 7, plus 12 percent unknown.

*** Sugimoto (1985) reports 7.2 percent of failures of drive systems. \(a\) is estimated to be 5 percent of these failures during programming.

**** It is expected that 1 out of 100 times the programmer will not notice an unexpected motion in slow speed within a dangerous situation.

The frequency of occurrence of accident scenario \(S_4\) is

\[(17) \quad p_4 = 0.005 \cdot 10 \cdot 0.01 + \frac{(1 \cdot 0.004)}{(2 \cdot 1000)} \cdot 10 \cdot 1 = 0.0005 + 0.00002 = 0.00052 \text{ accidents per robot per year}\]
This is about 5 times lower than the reference value of 0.003 of formula (7). The question is, how close are we really from such an accident? Is formula (17) representing the median value, or does it reflect some upper bound? Another question is, whether it is acceptable or not.

6 PREVENTION STRATEGIES

Looking back to formula (17) this gives the opportunity to define installation-specific solutions. Both types of initiating events appear to be of the same order of magnitude.

Well-trained programmers and off-robot preparations before the on-robot programming commences may reduce $p(\text{UMPR})$ substantially. Another solution may be a built-in control system, which registers deviations from the preset trajectory. One problem, however, is the possibility of distinguishing between what is a preset trajectory and what is a programmed trajectory. In fact, there are two types of unexpected motions, type $\text{UM}_1$ which is the result of an erroneous programming action that is not detected in time, and type $\text{UM}_2$ which is the result of an error in the computer program itself and not detected beforehand (the programmer missed that one, or it slipped into the program during the time of programming). Both types of unexpected motions can be dealt with separately in order to reduce $p(\text{UMPR})$ further.

A second man outside the robot area with his hand close to an emergency button may even reduce $p(\text{ftdIUM})$ further, albeit quite a boring task.

A reduction of the full speed error can only be achieved by improving the robot’s reliability with respect to this type of error (lower $\tau$ and, if possible, higher MTBF) or by introducing a safety system which responds to an off-normal speed increase. Such a safety system will always increase safety as the conditional probability of a fail-to-danger state given a full speed error is now approximated by unity. Such a system may consist of a positioner, a computer which calculates the actual speed and compares with a preset value, a transmitter which gives a signal to the power cut-off switch in cases the comparison is negative. Even if the reliabilities of these four elements are not that good (e.g. less than 1/100 hours) and the system operates quite safe when the safety system is tested before programming. If tested in the morning and used for the programming that day the probability would be

\[
p(\text{ftdFS}) = 4 \times (1/100 \times 8) = 0.32
\]

which is a factor of 3 improvement. It, however, requires a test before every programming action starts. Otherwise $p(\text{ftd/FS}) = 1$ again.
7 CONCLUSIONS

Suppose Carlsson's data were more or less representative for robot accidents in the early eighties, then tables 4 and 5 could indicate what the accident prone risk factors would have been (e.g. in the car manufacturing industries it would have been maintenance personnel and disturbances in the robot sequence). This indicates the contribution of the accident scenario concept in a risk assessment.

The safety policies implemented recently into a car manufacturing industry appear to be very effective in preventing accidents of operators. The results of the car manufacturer are in concordance with that. Special care is still necessary for maintenance workers and programmers as well as their relevant accident scenarios cannot be eliminated or reduced to a negligible level of risk in the deterministic approach. Whether the present safety policies should be extended for both groups is of course left over to the decision makers within the company. The presented example of a probabilistic approach to prevent accidents of programmers provides input data for such decisions.

8 REFERENCES

Bonney, M.C. and Y.F. Yong, Robot Safety, IFS(Publications) Ltd, UK, 1985


DIN, Vomorm DIN 19 250, Jan. 1989


JISHA. Prevention of industrial accidents due to industrial robots, Japan Industrial Safety and Health Association, December 1983


In this paper I shall give an introduction to the basics of the robotised terminal for container handling in Rotterdam.

In 1992 the new expansion of the Delta Terminal will be operational. Automatically Guided Vehicles (AGV's) will transport containers between the human controlled seaside cranes and the automated landside cranes. The working area of the AGV's is about 1200 meters long and 150 meters wide. All AGV's are connected through a radio data link to a central computer.

This central computer controls all the actions, which may concern a container move as well as an instruction to refuel or to go into preventive maintenance. A typical instruction could be the following:

"AGV number 16, move from your position through points A,B,C,D towards seaside crane 2, get container number EFGH 123456 and deliver this container through points I,J,K,L to landside crane number 12."

This sounds very simple, but the execution of this instruction is immensely complex. First of all the AGV has to navigate along a pre-determined path towards a well defined goal. It must be able to do so in bright sunlight, in darkness, in fog and in snow. Then it must dock under the crane with a precision of one inch, or the twist locks cannot be energized. On the way towards the crane it must be able to avoid obstacles and report them to the central computer. Also when two or more AGV's approach each other, hopefully in different lanes, their sensors must not interfere with each other.

We can choose between two different approaches for the central computer system that has the final responsibility for the movements. One may choose for intelligent AGV's that are able to cope with local problems or one may choose for non-intelligent AGV's that report everything they perceive and wait for an OK to proceed further. The advantage of the first approach is that the instructions can be simple and the communication with the central computer very restricted.

In order to give you an idea of the complexity of the terminal here are some numbers. On a yearly basis 500,000 moves have to be made. 8 seaside cranes and
25 landside cranes are required to move the containers from ship to stack. These movements are done with AGV's. 50 AGV's at a top speed of about 3 meters per second transport the containers.

In this paper I will first tell you about the navigation of the AGV's. Next I will speak about the sensors and the local intelligence, the communication with the central computer, and the prototyping.

1 NAVIGATION

The AGV's have to run in several fictitious pre-defined tracks over the terminal. They must be able to change their lane, take the next left etc. The question arises, must those tracks also be physically defined or is it better for flexibility to use more abstract means.

The most simple system I can think of is to paint white lines on the terminal, and have some optical device trace this line. This works excellently for a part of the year. However, when it snows this system will not work.

The passive induction system is a second contender. Here strips of conducting material are placed under the stones. At the AGV there is a magnetic coil that energizes the conducting material which will respond with eddy currents and the associated magnetic fields. These fields are measured and translated into lateral displacement. The main problem with this system is the distance because designers of the AGV want at least 25 cm clearing under the AGV. Add to this the size of the stones and the system has to perform at a distance of 40 cm. These systems have been realised at working distances of a few cm, but as the signal goes down in proportion to distance with the fourth-power this really poses a problem. Currently we have a system like this under test.

The active induction system is based on current carrying wires. Antennas pick up the signals and transform them into steering signals. The disadvantage of this system is that it is both inflexible and vulnerable because whenever a wire breaks one track is out of order. Tests are being done at the Delta terminal with this system. At this stage - the tests are still in progress - the results so far, however, are not very convincing.

The FROG system as it is being tested consists of the following elements: transponders which are placed under the pavement and antennas which energise the transponders, read out their code and fix their location. This could give a very flexible solution to the navigation system. Final tests will be performed soon.
The ultimate system would be a vision system where we have two CCD camera's in front of the AGV interpreting the landmarks in order to fix the position. So far we have, regretfully, not yet found a vision system that will perform under the climatic conditions of the Delta Terminal.

In addition to the systems described above we need devices that will help us to go from track to track or will help us to make 90 degree turns. We have looked at many systems, but so far we have not found the ideal solution. On the AGV's, odometers have been installed. These odometers measure the rotations of the wheel with a precision of about one cm. With odometers both on front wheels and rear wheels it is possible to get a reasonable degree of precision in addition to the other navigation systems. The main problem is sliding. If the AGV slides over some distance the error will be as big as the sliding distance. Therefore the navigation system has to be recalibrated at regular moments.

In addition to odometers one can also install a flux compass. These compasses are very precise (± 1 degree) and not very expensive. The main question is how this compass will interact with the mass of iron of the AGV and the container. What will happen if somebody ships a container full of magnets? Soon a flux compass will be tested on an AGV.

The most precise navigational instrument is the laser gyro. A commercial airplane equipped with laser gyro and inertial transducers can fly from Amsterdam to Los Angeles and be less than one mile off, a captain told me recently. However, the cost of the laser gyro with related electronics is regretfully prohibitive.

ECT is still in the phase of testing the various navigation systems, and so far it is not yet clear which system will be selected, as each system has its advantages and disadvantages. What emerges as essential is that each AGV must have several systems that combine their information to the best guess where the AGV is located. As rotational errors are by far the most dangerous, a compass is a necessity.

When docking, the AGV must navigate with a precision of 2.5 cm. This could be done with a laser beam on the AGV that is reflected by a mirror at a fixed location under the crane. Under the landside cranes which are at a known location this docking could also serve to recalibrate the navigation system.

2 SENSORS

The sensors of the AGV have one important function: they must prevent collisions of any kind. The AGV can move forward, backward, make a turn, and run crabwise. This means that the sensors must be on all sides of the AGV. The braking
distance at full speed, which is about 3 meters per second, is about 6 meters under dry conditions. Therefore the sensors which are mounted in front and in the back of the AGV must be able to perceive objects at at least this distance, preferably somewhat more like 10 to 12 meters.

Directly related to the problem of sensing objects is the question of passive or active sensing. Passive sensing is defined as sensing an object by means of energy reflected by the object. In many cases this can be very problematic, as even large surfaces under 45 degrees act as mirrors and do not reflect any energy back to the source. Therefore one thing is sure: all AGV's have to be coated with some material that reflects the energy in all directions. The type of coating depends on the wavelength of the emitted energy.

The ideal sensor would be an instrument that scans around the AGV and determines if there are any objects and if so, where. IBEO produces a laserscanner that scans an area of 270 degrees and measures the distance to objects in this field. The associated computer is programmed in such a way that parts of the field of view can be blinded. For instance, if such a scanner was placed at the right front corner of an AGV we could instruct the computer to look ahead for only 12 meters and to look aside for only one meter. I use the word "only" because the manufacturer claims that the scanner can "see" objects as far as 100 meters. When using two such scanners, one in front at the right side, one back at the left side, the complete area around the AGV is scanned. The scanner must still be tested for its performance in blizzards. A scan will be made ten times a second. During this period the AGV at top speed has moved 30 cm. The precision of the distance measurement is about 1 cm. Assuming that the scanner will work in fog and snowstorms, and assuming that we can protect it from clogging snow or ice, than we can speculate further. At top speed the inherent precision is of the order of 30 cm. As the lanes are 100 cm apart this is quite sufficient for safety. But, if we also place markers along the track, we can also navigate with this scanner. The layout of the terminal must be fed into the memory, and after one scan the AGV knows exactly where it is and on what course. In this way a sensor can be upgraded into a navigator. Two questions have still to be answered: can it really look through snow, and can the price compete with other systems?

Having discussed the most advanced sensor I would like to speak about the most simple one: the ultrasonic sensor. An ultrasonic sensor gives a short shriek after which it listens to reflections of the sound. The frequency of the acoustical wave can be anywhere between 40.000 Hz and several MegaHertz. Most commercial devices work at 40 kHz. As mentioned earlier, the ultrasonic detector measures the distance to the object, but it cannot give the exact location because of the angle of transmission and reception. This angle is proportional to the diameter of the transmitter divided by the frequency of the transmitted signal. There are ultrasonic
transducers that will work easily up to 20 meters with a beam angle of a few degrees. They could perform well for collision prevention, but what if an object, at a distance of 10 meters, 50 cm besides the track presents itself while a gale is blowing. The sound waves will be distorted by the wind, and the object will be detected as being in front of the AGV. Considerations of this kind have led us to the opinion that ultrasonic sensors can best be used for close detection of objects for a distance of a few meters in order to prevent maintenance people being hit by AGV's.

2.1 Radar detectors are of a different kind

We have extensively tested the MX-5 sensor from Siemens. It emits electromagnetic waves of a frequency of 9 Gigahertz, which puts the wavelength at about 3 cm. This frequency will look through almost everything non-metallic. In fact it could detect movements right through a flagstone. It works on the principle of Doppler shift and it only detects moving objects. As the beam is very wide, the only significance of a signal from this sensor is that there is something somewhere in the field of radiation, but the whereabouts are unknown. We designed special electronics to lower the detection frequency of the Doppler output, such that even at the trailing speed, objects would be detected. With the use of a parabolic reflector the beam can be narrowed to a few degrees.

This sensor needs careful consideration. It is ideal in the sense that it is electronically simple, looks through ice and snow, can be sealed easily and performs very reliable. However, it gives no clue to the distance to objects. This sensor can also be used as an odometer. When installed at 45 degrees to the ground it will measure the distance covered by a click every 3 cm. Unlike the wheel-mounted odometer, this odometer will also measure the distance while sliding. One problem remains: at very low speeds the output will give no signals due to the inherent noise.

2.2 Infrared array

The infrared array was developed for the AGV. It consists of two transmitters that transmit a strong beam of modulated infrared light with an angle of about 8 degrees and two receivers with the same aperture. There are 16 emitted beams and 16 receiver beams. The array is computer-controlled by a local computer. It can initialise a scan of 16 beams. Whenever there is an object on the cross-point of transmitter and receiver, and when reflected energy is sufficient, the output of the receiver will be set to one. Ten consecutive sweeps will be made in about 2 milliseconds and the results of the outputs will be added to cancel out random
effects. If there is an object in the cross-point of beam 5 and receptor 7 the count of hits should be 10. This information will be transmitted to the AGV computer.

Preliminary tests show that a man in a dark suit can be detected at a distance of about 10 meters. These tests also showed that a good reflecting surface like aluminium cannot be detected at 3 meters, while plywood seems to reflect very well. We are assuming that this array system will be able to penetrate through mist and snow.

It is appropriate to return to the reflecting properties of the AGV, because this is a main cause of concern. The best reflector that reflects diffuse energy is probably something like rough sandpaper. This reflector will reflect laser beams as well as radar beams. But under icy conditions, the structure will be filled with ice and again we have an invisible object. Therefore it seems imperative that the structured reflector be heated when necessary.

2.3 Local intelligence

Local intelligence is completely related to the sensing ability of the AGV. A blind and deaf AGV can be provided with the best algorithms and computers, but it will stay where it is, because it does not know where to go. Let us take the laser scanner.

This sensor system can be made very intelligent. It is even conceivable that the central computer will issue orders like:

"AGV 14, find seaside crane 4 and take a container. Drive carefully, be happy, don't worry",

whereupon the AGV takes off, across the terminal, defines the best way to get to the crane, and docks there. After reception of the container, the AGV takes it to landside crane 4, because this one is nearby, given the traffic situation and stacking is random anyway.

In reverse, in this MSS system, which is majority from sea to sea, an instruction could be like:

"AGV 14 go to landside crane 7 and get container ABCD123456 and bring it over to seaside crane 3".

This already is much more difficult. How is the AGV to know when it is its turn to load the container? The AGV is unaware of the sequence of delivery of containers at landside crane 7, and it is not able to distinguish one container from the other.
Maybe at some date in the future it will also be able to read a code on the container and line up in the proper sequence. All this, however, is wishful thinking at the present time.

ECT has chosen a very strict control of the AGV’s, with a minimum of local intelligence. Local intelligence should be like the central nervous system. It only overrules orders when it has to. When instructed to drive another 10 meters, it will drive another 10 meters unless the sensors detect an obstacle. Then it will stop, regardless of the order issued.

2.4 Safety conditions

All mechanical functions of the AGV have built-in watch dogs that have to bark at regular intervals. If the barking stops, so will the AGV.

It would be nice to have a similar system both for the navigation system and the sensors. For instance, if the passive induction system will be selected for the navigation over the terminal, there is a problem. If the coils are right over the metal conductor, the output will be zero, no steering is needed. But if the main coil burns out, the signal will be zero as well. Also, if no conducting material is underneath, the signal will be zero.

It needs a lot of electronics to guard for these conditions. If the laser scanner scans nothing, this means that the laser has burnt out, and consequently the AGV should grind to a halt. Suppose now that one or more beams of the infra-red scanner burn out. This will go undetected, because in order to prove that the system is working, there has to be an obstacle. Therefore it is advisable to test all open loop detectors at docking places where the response is available.

3 COMMUNICATION

In order to control 50 dinosaur-like AGV’s, one needs a strong voice and a wide vocabulary. In the current concept of control the AGV’s have to report their position every 3.3 seconds. Even when they do not move, this report is required to check the communication channel. The length of the message is 5 bytes, the acknowledgement is 3 bytes long. What the AGV is reporting is its location and its orientation. The response of the central computer - PCS (process control system) - is to give a go/no go signal for the next 10 meters and the required speed.

This also explains the repetition rate of the communication. At 3 meters per second the AGV covers 10 meters in 3.3 seconds. The PCS processes the information and
it must be able to do so in about one second. About ten times per hour the PCS issues a new order for an AGV. This order is 21 bytes long. The response is 4 bytes long. There are also messages concerning the status of the AGV such as vehicle statistics, yellow and red status of the mechanical, and electrical equipment, start-up and stop procedures. Extensive simulations on the channel capacity for the AGV’s have shown that it must be possible to control 7 to 8 AGV’s on one radio channel of 3600 baud with 9 bits out of 16 for redundancy. The radio system under test is produced by Autophon. There is a back up channel that can be used when one of the other channels goes down. Back up procedures have been defined in such a way that, if a channel goes down, all users automatically switch to another channel. Then the disabled channel is no longer a back up for other users.

The PCS is connected to the radio channels that ECT has been given by the PTT. This is a rather complex situation. My personal preference would be to use a modern 64 kilo baud channel which is able to serve all AGV’s.

3.1 There is an alternative: infrared communications

Several years ago a vertical infrared communication system was installed at the home terminal. The speed of this system is 100,000 baud with a redundancy of 50%. More recently a horizontal infrared communication system was developed with the idea of being able to communicate to the central computer from anywhere. This HIR system could cover a distance of 140 meters under good climatic conditions. In fog or snow, measurements indicate that the working distance is limited to 50 meters. One could install an infrared communicator every 50 meters on the terminal, the maximum distance to every AGV would be at most 25 meters. As the AGV’s are supposed to drive very precisely, these posts do not present themselves as collision objects but rather as landmarks when referring to the IBEO system for navigation. Each AGV has to be equipped with two sender/receivers, one in front and one in the back.

4 PROTOTYPING

In 1990 eight AGV’s will be available for an intensive prototyping test. During this prototyping all system functions will be looked at through a magnifying glass. The observers are separated from the AGV’s by a safe distance, they watch the movements of the AGV’s, diagnose any shortcomings of the system, and try to improve the performance.

There are three main areas of tests. These are:

- efficiency
- reliability
- safety
With respect to the safety aspects one general rule has been applied at all levels of the design of the AGV. Whenever a device is unenergised, it will fall into the safe status. For instance, if the energy circuit on the brakes is down, the AGV will brake. If the radio data-link is down, the AGV will stop. What will happen if an AGV is hit by lightning is unforeseeable but chance has to play a very weird role if anything serious would occur.

The advantage of doing a prototyping test is that unforeseeable situations may occur and be dealt with in an adequate manner, while the risk of operational damage is low. The prototyping test serves to test the software and to modify or improve the software - maybe also the hardware - whenever needed. As these tests will take place next year it is premature to expand on them.
1 INTRODUCTION

The multi-disciplinary field of computer vision is concerned with the methodology and the implementation of algorithms for the manipulation of images, for the extraction of information from the images, and, eventually, for the interpretation of the scene which is represented by the digital images. It includes image sensing, image digitization, image filtering and segmentation, measurement of numerical or structural features, pattern recognition and, eventually, scene interpretation. Disciplines involved range from signal theory to artificial intelligence.

In the industrial sector the promising applications are in the areas of materials handling, inspection, continuous-process control and surveillance and particularly in the field of robot vision. The latter case may include guidance of autonomously moving vehicles, but here we will concentrate on vision for flexible assembly cells in the context of the SPIN/FLAIR-2(DIAC) project at Delft University of Technology.

In a flexible assembly cell, vision can be used for the following tasks:
- Detection, identification and localization of parts offered to the cell;
- Establishing a map of avoidable obstacles in the cells work areas;
- Checking and inspection of the result of an assembly step;
- Recovery from errors by a possibly slow but careful investigation of unexpected situations in case of detected exceptions.

The first two points have a direct relationship with the issue of safety in the assembly cell.

It is quite obvious that object identification and scene interpretation require the presence of models and the incorporation of knowledge into the vision system. However, it should be noted that the use of models and knowledge will also greatly facilitate the earlier processing steps.
1.1 Data acquisition in 2-D, 3-D and 2.5-D

In terms of the perception of objects and landscapes composed of objects, 2-Dimensional (2-D), 3-Dimensional and an intermediate level of 2.5-Dimensional (2.5-D) vision systems may be characterized as follows:

- Objects perceived in 2-D will result in an image which represents the face of that object. The data representation of the image is usually a 2-D integer array, where each array value (picture element or pixel) is a grey value or light intensity point. An example is a digitized image from a video camera.

- Objects perceived in 3-D will result in an image which represents the object as a pile of cross sections. The data representation is usually a 3-D integer array, where each array value is a grey value or intensity point. In case of solid objects the integer array may reduce to a binary array (object / no-object). Each 2-D subarray is a cross section. Examples are the images obtained with computer tomography (CT scan). Full 3-D image acquisition is hardly feasible in the context of robot vision, and hardly ever necessary.

- Objects or landscapes perceived in 2.5-D will result in an image which represents a depth map of that object or landscape. The data representation of the pixels is a 2-D integer array, where each array value is a distance value or depth point. Such 2.5-D images can, e.g., be computed from 2-D stereo pairs. The 2.5-D perception of objects may be useful in situations where the third dimension of at least some points is assumed to be fixed, as is the case with objects in a stable position on a flat table. Range data is also important in assembly checking and is necessary in robot collision avoidance. Hence, 2.5-D vision is extremely important in many robot vision applications.

1.2 Computer vision pipeline

Sensors are often more or less general purpose and offer more information than necessary for certain specific tasks. Tailoring the data is usually more cost effective than tailoring the sensor, especially when more than one task has to be performed with the same sensor. Each vision task requires a number of processing steps which results in a pipeline of procedures, as illustrated in figure 1.

Within each processing step, essential information for the required task is extracted and often the image data are transformed into a more compact representation suitable for the next processing step. In general, the required processing time decreases when the image description becomes more compact. This leads to the objective to obtain compact data representations as early as possible in the pipeline.
Different tasks may share pieces of the processing pipeline, i.e. some procedures and/or representations. The various representations depend on the data elements used in the consecutive processing steps, their topology or interrelations and the implementation tools used to represent and process them.

1.3 Checking the results by using models

However complex, the world within an assembly cell is quite simple in comparison with the universe. Most, if not all, of the objects which are present in the scene may assumed to be known in the form of models (statistical, structural) and in many cases the approximate positions of these objects within the scene can be modeled as well. This leads to the observation that we may frequently start with some hypothesis about the observed object or scene. In other words, the general problem of pattern recognition and scene interpretation may be viewed as a hypothesis
checking in many situations within the assembly cell. This can be achieved by computing, in some way, the similarity between the observation data and one or more prestored models.

A simple case is the check on the dimensions of an object. The matching of observation and model degenerates to a check whether the measured values are within prespecified tolerances with respect to the modeled values. Basically, this corresponds with the feature space approach of statistical pattern recognition.

For many problems, a structural approach is more appropriate. In this case, inexact graph matching has proven to be a good tool. The basic idea is that a graph is constructed from the observed data, which is then compared with a prestored model graph. In principle, the matching can be done in 3-D (3-D/3-D matching). A more moderate and attractive approach is the matching of 2-D images with a 3-D model, which implies the generation of 2D model projections or vice versa.

2 VISION SYSTEMS FOR FLEXIBLE ASSEMBLY CELLS

2.1 Vision tasks and vision systems

The Delft Intelligent Assembly Cell will contain two concurrent or cooperating robots, each with its own work area as well as with one common work field. Parts are fed into the system by a feeder device. 2.5-D vision systems may play a role in at least four tasks:

1. Detection, identification and localization of large parts offered to the cell by its environment.

2. Provide all necessary information to perform collision avoidance in case of coarse robot movements, both between robots and between robots and objects.

3. Checking and inspection of the result of a just performed assembly step.

4. In case of an exception, a scrupulous slow investigation and analysis of the situation in the scene may be necessary in order to facilitate error recovery.

For task 1, knowledge of the expected objects is present. Large parts are presumably delivered in a more or less fixed orientation on pallets by means of an automatically guided transport system. The parts do not overlap. The pallets may contain different types of parts at prescribed positions within the pallet. Smaller parts may be delivered in boxes containing one type only. These parts may overlap.
Tools used by the robot, like grippers, screwdrivers or even inspection camera’s, will be handled as known objects. Unidentified objects will give an exception situation. The vision system should be able to detect an object on a fixed background scene, identify or reject the object, derive its position and orientation, retrieve necessary grasp data from a data base, check the possibility for grasping and return the order of grasping to the calling system. The typical field of view (FOV) should be about .6 x .6 x .3 m. The typical resolution should be about 1 mm$^3$. In DIAC, a stereo vision system using two (or more) video cameras will be used for this task. If a point in 3-D can be observed in both 2-D images of a stereo pair, its position in 3-D can be computed from the two 2-D positions and the camera geometry by means of triangulation. This requires the matching of the left and right images. In general, straightforward gray value correlation techniques are less attractive than matching on the basis of detected characteristic features like edges, corners, faces, etc. This can be accomplished by using graph matching techniques. From each image a 2-D graph is derived. One can match the two 2-D graphs against each other (2-D/2-D matching) or each of the 2-D graphs against a 3-D model (2-D/3-D matching). In the latter case one can map the 2-D images on the 3-D model or project the model on the 2-D images. On the basis of resolution, depth of field, sensitivity and accuracy stereo vision apparently is a good solution for task 1 as defined in the previous section (sensing, detection, identification and localization of parts). The DIAC stereo vision system will be described in more detail below.

Task 2, the collision avoidance is of particular importance in the common work space of the robots and concerns both collisions between a robot and some object as well as between robots. The collision avoidance problem may be solved in a rather coarsely quantized space, allowing fast coarse detection of objects by the vision system. A bookkeeping of the world, combined with expected results of requested robot and object moves, are checked against the observed images. Small deviations will update the system (adaptation to the real world), large deviations will give an exception situation including the location of the deviation. The typical field of view is about 1.5 x 1.5 m. The typical resolution should be about 10 mm$^3$. In DIAC, time sequential binary space encoding, using a CCD camera and LCD based lighting will be used for task 2. In this technique various binary lighting patterns are projected on the scene, sequentially in time. Each of the reflection images is digitized and stored. The basic idea is to use the geometric pattern in the illumination to help extract geometric information about the scene. A combination of 14 binary patterns is required to obtain about 7 bit depth resolution. Due to resolution, depth of field, etc., this appears to be an attractive technique for task 2 as defined above (collision avoidance). In DIAC the illumination patterns will be projected by means of a commercial slide projector with fixed LCD. In task 3, the assembly inspection, the task is to inspect whether the mounting or fixing of parts together was performed correctly. A sensor with high resolution is necessary here.
Moreover the high resolution must not lead to slow measurements; a fast sensor is profitable. Matching of measurement results with prestored requirements in a database may be necessary. Dependent on the situation, deviations will give a reject or an adaptation of the database. The typical measurement area will be at most (.25 x .25 x .25) m. The typical resolution with this FOV should be about .5 mm³. Measurements for diameters etc. with a FOV of about (.1 x .1 x .1) m should be performed in about 0.1 mm³, the accuracy of the robot. In DIAC, active laser slit lighting and triangulation, using a PSD array and laser LEDS will be used for task 3. Slit lighting is another example of the use of structured light. A single plane of light is projected onto the scene, which causes a stripe of light in the scene. Only the illuminated part will be sensed by the position sensitive device. Light source and sensor are arranged geometrically like in stereo vision, and triangulation can be used to compute the 3-D position of any image point visible as part of a stripe. A depth resolution of 8 to 10 bits can be obtained at a typical field of view of 30 cm. The characteristics of this system make it an attractive solution for task 3 as defined above (checking and inspection of results of assembly steps).

Task 4, the exception handling / error recovery is very speculative. Exceptional situations may occur in performing tasks 1 and 2, especially. An example is when a robot has dropped an object somewhere in the scene. In this case the exception handling is a true extension of the collision avoidance problem. Techniques used in task 1 may be used to identify unknown objects in order to enable their removal. Adequate handling of totally unknown objects is at the moment out of the question. Task 4 can be split in tasks conforming to tasks 2, 1 and 3 and can thus be accomplished by the systems described above.

Figure 2: stereo vision system for object identification
2.2 The diac stereo vision system

In DIAC, the stereo vision system will be implemented with two or more (color) video cameras directly above the part feeder location as illustrated in figure 2). Figure 3 shows the architecture of the vision pipeline used for the stereo-vision system. The pipeline steps from camera to graph matching are implemented in duplex.

- Each pipeline starts with a (color) CCD-camera acquisition system. Optionally, a camera control system might be used for zooming, panning and tilting possibilities. This could be used for fast mutual calibration of the two cameras or for the vision function itself.
- If necessary grey value filtering will be applied, e.g. to remove white peaks (specular reflections) and/or shading.
- The resulting image is segmented into objects and characteristic features of the objects are derived.
- The two pipelines merge in the graph matching phase, where the left and right projections of the 3-D scene are matched with wire frame models of objects derived from the CAD database. The objects in the scene are supposed to be designed with this database.
- The stereo-vision pipeline is supervised by an intelligent controller which can, to a certain extent, reason about the requested task. This issue will be discussed in some more detail in the sequel.

![Figure 3: the architecture of the stereo vision pipeline](image)

3 SEGMENTATION TOOLS

An important step in the image analysis process is the segmentation of the image into regions which are homogeneous or coherent according to some predefined
criteria. The result is thus a map of the detected regions, e.g. various object faces and a background region. Usually, the following families of segmentation methods are distinguished:

- Parallel region oriented segmentation: Thresholding
- Parallel edge oriented segmentation: Edge detection
- Sequential region oriented: Region growing
- Sequential edge oriented: Edge tracing

In general, it can be stated that if the vision tasks will be performed in software, only grey value thresholding will be fast enough for robot vision. This segmentation technique can be applied in some very simple cases only. An example is the 2-D case with sufficient contrast and low noise level (object silhouette recognition).

For the recognition of 3-D objects using graph matching techniques, edge oriented models of objects and edge detection techniques seem more suitable. Parallel methods are somewhat less useful here in the sense that much time is lost in processing parts of images that later on appear to have no coherence at all. Hence, in the robot vision literature much attention is paid to the subject of edge tracing, using clearly defined starting points of objects. Straightforward implementation of these methods is however too slow. The research on segmentation techniques for the DIAC project is focussed on two tracks: the multi resolution approach and the knowledge based approach.

The multi resolution approach is based on the idea of planning in the sense of Kelly [6]: Search in the low resolution version of the image for interesting parts of the image to define regions of interest. Within these relatively small regions of interest one can allow the use of relatively complicated methods. Parallelism per region of interest is a suitable strategy if the (hardware) facilities allow this. On a higher level, the results have to be combined but involve less data. Within the multi resolution approach, split and merge methods and pyramid methods are applicable and seem without much problems extendable to the 3-D case.

The knowledge based approach is based on the observation that information from prestore models or from previously processed data can be used to guide the segmentation. In the former case, a dramatic reduction in processing time can be achieved by restricting the time-consuming processing of grey level images to relatively small regions of interest. In the latter case one can consider information from lower resolution images (multi resolution approach), from other projections (stereo vision) or from another sample in time (moving belt). If this information is exploited intensively, again a substantial reduction of processing time can be obtained.
In DIAC, the developments are towards a modular segmentation mechanism controlled by a local intelligent controller using strategy rules, that can handle actual situations in a flexible way. The accent will lay on grey value / color thresholding (trivial) and the tracing of edges using planning.

4 INTELLIGENT CONTROL

Knowledge-based controllers can presumably be used in both the stereo vision system and in the structured light system. Task flexibility, variable process time, variable resolution and the selection of regions of interest might be performed with such a controller for the stereo vision system. The controller might update the parameters of the pipeline process steps and act thus as a feedback mechanism to optimize the pipeline [2], [8]. The role of an expert system controller in the structured light pipeline (collision avoidance) would be to form a feedback mechanism to handle small differences between observed world and expected world. Either the database interface, the vision system or the robot move generator might be tuned by this controller.

In [2] a flexible and intelligent 2-D binary measurement system is described. In this system, the processing modules of the vision pipeline are implemented in a modified blackboard structure which also contains modules for feature selection and learning. In the blackboard structure, the controller and the interface to the robot cell play a dominant role. They perform four distinct types of operation: learning, initialization, execution and communication. In the learning stage, all possible measurements of all presented objects are stored in the knowledge base (Long Term Memory). This is done completely off-line. In the initialization stage, the actual task is defined. The controller activates the feature selection module which selects the subset of features which is optimal for the current task and which can be measured within externally defined processing time limits. Depending on the selected features, the controller defines the processing path required by the current task because various features may require different image processing steps.

The developments for the stereo vision object identification system are along the following lines. Many objects will have a limited number of stable positions when placed on a flat surface and from the 3-D model some typical views can be derived. This is done off-line and the results are stored in a database. In the initialization mode, the current task is defined. Usually, only a subset of the pre-learned objects play a role in the current task and a task-adapted generalized feature selection will be performed. The generalization is that both numerical features and graphs will be considered. The design of such a generalized feature selection module is the object of current research. For each of the 'active' objects some generalized features will
be selected with the best discrimination power with respect to the other 'active' objects. The results are stored in the database, and are used by the controller to define the processing paths for the task execution stage.

In order to gain speed in the task execution stage, the routine processing is based on assumption that the presumed object is present indeed. This implies that model knowledge can be used, even to the extent of defining regions of interest in the segmentation stage. This routine processing continues as long as all evidence from the observations supports the assumptions. If the observations are contradictory to the assumptions, the controller switches to exception processing, in which case model knowledge can hardly be used to guide the processing.

The stereo vision system will be extended with a remotely positioned top-view camera with telelense. Secondly, the system may have up to four cameras closely arranged around the scene, and the image from any of these cameras can be selected for analysis. Due to the telelense of the top-view camera, this image contains almost no 3-D information at all, but will be used to obtain some very simple features (image histogram, object size after thresholding) for a first check about the hypothesized object and to obtain coarse estimates of its position and pose. This information is used by the controller to derive from the database which close-view camera is to be selected and the corresponding view-model is used in the model-guided analysis of the image. The processing is restricted to obtain the necessary evidence for object verification. Finally, the model information and a few geometric features of the object are used to accurately compute object position and orientation.

5 Vision and Safety

The role of vision as regarding safety aspects in robot systems should not be overestimated. Certainly, there are safety aspects involved in the correct identification and localization of objects in the sense that erroneous and thus potentially dangerous robot actions are prevented. The same argument holds for the role of vision in path planning and collision avoidance. However, the role of vision systems for general surveillance of the robot work space is very limited. The processing of large grey-value images and especially the automatic interpretation of complex scenes in search of potentially dangerous situations is, in general, much too time-consuming to provide a real-time feedback to the robot controllers. In the above, we have described the use of a priori knowledge and especially the issue of restricting the processing to small regions of interest within the images in order to meet the time constraints of the assembly process. Such ideas are contradictory to the task of general surveillance for safety purposes. Solving such task by means of vision systems, if at all possible in the
environments described here, would require computing power and special purpose hardware which is in no way economically acceptable.

ACKNOWLEDGMENTS

The contributions of D.J.W. Bierhuizen, J. Buurman, R.P.W. Duin, M.A. Kraayveld, P.W. Verbeek, and I.T. Young to various aspects of the work reported here, are gratefully acknowledged.

REFERENCES AND LITERATURE

Books:


Articles and reports:


Intelligence, Vol. 6 pp 397-409.


Abstract

The use of robots is growing from small scale applications in a single vendor environment to medium scale applications in a multi-vendor environment. With the release of MAP 3.0 the choice between a multi-vendor network system and the integration of multiple single vendor products has become relevant now. This choice may influence the safety and reliability of the total system. A short overview on the historical and technical issues of MAP is presented. Advantages and disadvantages are evaluated in relation to safety, reliability and costs. This is done for the installation of new systems and for the expansion of existing systems.

Keywords:
CIM, MAP, safety, reliability.

1 INTRODUCTION

In Computer Integrated Manufacturing (CIM) communication is considered to be a key issue to success. Among the many (single-) vendor solutions to the communication problem there is an alternative which is supported by many vendors: the Manufacturing Automation Protocol (MAP). MAP is the standard communications solution to flexible manufacturing systems proposed by the International Standardization Organization (ISO) and recognized worldwide as a solution to the communication problem in computer integrated manufacturing.

This paper addresses the issues of safety and reliability of MAP based solutions as opposed to the integration of systems from different suppliers (multi-vendor) on a per system basis.

The structure of this paper is as follows. After this section the integration of multi-vendor equipment is discussed in Section 2. In Section 3 a technical introduction to
MAP is given. In Section 4 relevant issues on safety and reliability of control systems in a multi-vendor environment are discussed. The migration of existing facilities to a MAP based solution is elaborated on in Section 5. Section 6 presents the results of cost evaluations done within TNO. Section 6 recommends on how MAP can be introduced in existing production plants. TNO's involvement in MAP based solutions is highlighted in Section 7, where in Section 8 an approach for how to start with MAP is given. Finally in Section 9 the conclusions are presented.

2 COMPUTER INTEGRATED MANUFACTURING IN AN MULTI-VENDOR ENVIRONMENT

2.1 Networking as the backbone for computer integrated manufacturing

It is already for a number of years now that networks are not only used at the enterprise level but are also found on the factory floor. In most cases networks for manufacturing systems are based on products from suppliers of control systems like Honeywell, Allen-Bradley or Siemens, or based on products from suppliers of general purpose computers, like Digital Equipment Corporation (DEC).

Some of the advantages of using networks in the manufacturing environment are the following:

- correct data: networking software incorporates extensive facilities for correct data transfer,
- simpler device controllers: the speed of the available networks decreases the need for disk-based controllers as programs and data can be down-loaded via the network,
- version control: via the network the control programs can be distributed from a central point, which strongly reduces version control problems,
- more rapid changes in equipment parameters or control programs: via the network changes can be made relatively easy and in a controlled way.

It may be clear that these aspects also influence the safety and reliability of the production environment.

Due to the increasing complexity of manufacturing systems, it is getting more and more difficult to find a single-vendor solution to computer integrated manufacturing. Rather than integrating several products on a per system basis, a far more flexible integration is possible if it is based on network standards. This alternative is not only to be considered by the Ford, General Motors and Volvo manufacturers, but also by medium size and small size manufacturers. Murphy's law also holds for these smaller manufacturing environments: if you selected three
new types of machines for your manufacturing system it is likely they contain control systems of three different suppliers.

2.2 Networking in a multi-vendor environment: MAP

By the end of the seventies General Motors had installed approximately:
- 20,000 programmable controllers,
- 2,000 robots,
- 40,000 intelligent devices.

As only 15% of these devices could communicate outside their local production cell, a major expenditure would be necessary to bring these production environments together to form a computer integrated manufacturing system. To tackle this cost problem General Motors formed together with its suppliers a Task Force in order to develop a specification for an independent computer network protocol: the Manufacturing Application Protocol (MAP). In a relative short period of time subsequent versions were released, leading to MAP version 3.0 (MAP 3.0) in June '87. This last version has been frozen for six years and only functionality may be added which is compatible with MAP 3.0. To prove that MAP 3.0 was not only a specification, the Enterprise Networking Event was held in June '88 (ENE '88) in Baltimore.

At this event both manufacturers and suppliers cooperated in the realization of a number of demonstration plants where the use of MAP and TOP (MAP's equivalent for the technical office) was demonstrated.

In the following section, MAP is discussed in more detail. In particular, its relation to the OSI reference model for networking will be presented.

3 OSI REFERENCE MODEL AND MAP

3.1 OSI reference model

In order to enable exchange of data between a variety of computer systems the International Standardization Organization (ISO) has specified a number of standards based on a seven layer architecture, see Fig. 1, referred to as the Open Systems Interconnection (OSI) reference model. In this architecture each layer provides an independent, supportive, functional level [1], [2]. In effect, the total functionality of this architecture allows for the exchange of data between applications at different nodes in the network.
Considerations of applying Communication Standards for Robot Control

A. van Buuren

<table>
<thead>
<tr>
<th>layer</th>
<th>name</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>application</td>
<td>provides standard services for applications running on a computer in the network</td>
</tr>
<tr>
<td>6</td>
<td>presentation</td>
<td>provides the translation to and from the standardized format used within the network</td>
</tr>
<tr>
<td>5</td>
<td>session</td>
<td>provides the control structure for dialogues between two applications</td>
</tr>
<tr>
<td>4</td>
<td>transport</td>
<td>provides transparent reliable data transfer between two end nodes</td>
</tr>
<tr>
<td>3</td>
<td>network</td>
<td>provides the routing of information packages through the network</td>
</tr>
<tr>
<td>2</td>
<td>data link</td>
<td>provides an error free data exchange between two nodes in the network</td>
</tr>
<tr>
<td>1</td>
<td>physical</td>
<td>provides the transmission of raw bits over the physical channel</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>layer</th>
<th>name</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>application</td>
<td>provides standard services for applications running on a computer in the network</td>
</tr>
<tr>
<td>6</td>
<td>presentation</td>
<td>provides the translation to and from the standardized format used within the network</td>
</tr>
<tr>
<td>5</td>
<td>session</td>
<td>provides the control structure for dialogues between two applications</td>
</tr>
<tr>
<td>4</td>
<td>transport</td>
<td>provides transparent reliable data transfer between two end nodes</td>
</tr>
<tr>
<td>3</td>
<td>network</td>
<td>provides the routing of information packages through the network</td>
</tr>
<tr>
<td>2</td>
<td>data link</td>
<td>provides an error free data exchange between two nodes in the network</td>
</tr>
<tr>
<td>1</td>
<td>physical</td>
<td>provides the transmission of raw bits over the physical channel</td>
</tr>
</tbody>
</table>

Figure 1: OSI reference model

The mutual independence between the different layers allows for flexibility in communications between systems. For example, at the lower layers different protocols and/or transmission media such as CSMA/CD (Ethernet), Token bus (MAP), Token ring (IBM) or X.25 (PTT's DataNet 1), can be used without affecting any of the higher layers.

The functionality provided by these standards has its price. In Fig. 2 the overhead, specified in octets, is given for the different layers.

<table>
<thead>
<tr>
<th>layer</th>
<th>overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>no additional overhead</td>
</tr>
<tr>
<td>Presentation</td>
<td>no additional overhead</td>
</tr>
<tr>
<td>Session</td>
<td>2</td>
</tr>
<tr>
<td>Transport</td>
<td>5</td>
</tr>
<tr>
<td>Network</td>
<td>35</td>
</tr>
<tr>
<td>Data Link</td>
<td>26</td>
</tr>
<tr>
<td>Total</td>
<td>68</td>
</tr>
</tbody>
</table>

Figure 2: message overhead

This means that messages increase with about 68 octets. When a transmission medium with 5 MB is used, an overhead of 0.107 ms is required for each package, neglecting the processing time within each layer. This imposes a speed limitation in the use of full functional networking protocols for manufacturing applications. A limit which should be considered before implementation decisions are being made.
In the early days of the OSI model only the layers 1-6 were specified, the specification of layer 7 was left to the designer. At present a number of specifications for (parts of) the application layer are filled in for, amongst others, the following standards:

- Manufacturing Automation Protocol (MAP),
- Technical and Office Protocol (TOP),
- X.400.

### 3.2 Manufacturing Automation Protocol (MAP)

In addition to the existing specification of layer 1-6, MAP provides additional specifications for the layers 1 and 7, as shown in Fig. 3. For the physical layer a token-passing bus on 10 MB broad band or a 5 MB carrier band is specified. Additional specification for the application layer, in particular suitable for manufacturing applications, is provided by the Manufacturing Message System (MMS). MMS itself makes use of the Association Control Service Element which is also used for a number of other application oriented standard protocols.

<table>
<thead>
<tr>
<th>Application layer</th>
<th>application interfaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO MMS</td>
<td></td>
</tr>
<tr>
<td>ISO ACSE</td>
<td></td>
</tr>
<tr>
<td>Presentation layer</td>
<td>ISO presentation kernel</td>
</tr>
<tr>
<td>Session layer</td>
<td>ISO session kernel</td>
</tr>
<tr>
<td>Transport layer</td>
<td>ISO transport class IV</td>
</tr>
<tr>
<td>Data Link layer</td>
<td>ISO MAC and LLC</td>
</tr>
<tr>
<td>Physical layer</td>
<td>Token-passing bus on Broadband (10 MB) or Carrierband (5 MB)</td>
</tr>
</tbody>
</table>

**Figure 3: MAP's use of ISO standards**

MMS provides the framework for manufacturing application programs. In addition to MMS a number of so called companion standards are specified which provide guidelines for using MMS for certain types of manufacturing equipment like robots, NC-machines, etc.

### 3.3 Advantages in using standards

The advantages of using MAP for computer integrated manufacturing are the following:
Considerations of applying Communication Standards for Robot Control

- uniform functionality
- conformance testing
- multi-vendor environment

The uniform functionality provided for a wide range of manufacturing equipment eliminates the need for extensive education of development- and maintenance staff. In particular the maintenance, modifications, and additions can be made relatively easy.

For MAP extensive conformance testing systems have been developed, where tests can be performed by independent organizations [3]. This certification procedure provides a high grade of inter-operatability between products of different suppliers at the MMS level.

The availability of MAP products from different vendors makes MAP especially attractive for a multi-vendor environment.

4 SAFETY AND THE RELIABILITY OF SOFTWARE

In Section 2.1 a number of aspects related to safety and reliability have been mentioned. These aspects were related to the use of networks as the backbone for computer integrated manufacturing. In addition to this the following advantages relate in particular to MAP:

- conformance testing,
- integration,
- common functionality,
- modifiability.

Conformance testing provides guarantees that supplied software will function at application level as specified. This eliminates the need for tailor-made communications software, or even worse, special drivers. Manufacturing equipment with a MAP interface can be accessed from a computer on the network at the MMS level. Only an application program is required to collect the information from this equipment. Another effect of conformance testing is that for new MAP products, compared to existing single-vendor products, a comparable reliability (or even higher) may be expected.

Test periods after installation for thorough integration tests are expected to decrease due to the intensive conformance tests a MAP product has to go through before it is released. Apart from shortening expensive test periods, also the risk of accidents due to unexpected behavior of the system will decrease. The validation of
this statement has been done at the ENE '88, where a number of demonstration production plants were built and integrated within the 5 days available before the start of ENE '88.

Due to the common functionality similar data from different types of manufacturing equipment will be available for the operator of the manufacturing system. This will result in a common control structure for a wide variety of manufacturing equipment, decreasing the number of possible operational errors.

Just as the operator, also the designer who has to incorporate modifications into an existing system will benefit from using MMS. He will be able to maintain a wider range of equipment not being bothered by higher or lower level communications protocols, but just with one application protocol: MMS. The expected decrease in testing time has already been explained above.

The above shows that the use of MAP for flexible manufacturing systems introduces additional benefits on top of the ones already applicable for the use of networks for computer integrated manufacturing.

5 GATEWAYS: INTEGRATING EXISTING FACILITIES AND NEW FACILITIES

In many cases where the choice for MAP becomes relevant, there will be an existing plant using a different network or no network at all. For these cases the following three solutions are possible to integrate existing facilities into MAP-based networks:

- Small grain solutions,
- Medium grain solutions, and
- Course grain solutions.

For the first category a small scale gateway, e.g. PC-based, is assigned to individual manufacturing equipment, performing the translation of the MAP protocol to the dedicated protocol used for that particular type of manufacturing equipment. Information available may be limited due to the limited functionality available in the original equipment.

The second category requires a medium scale gateway which may be either connected to the existing network, or to the existing cell or area controller. In this case too, information can be obtained only within the cell or area controller.

The third category is formed by gateways developed for existing network based control systems.
The choice between the first two solutions has to be determined for each specific case. In general, one may expect that as more different manufacturing equipment has to be connected, it is more likely that a medium grain solution will be cost effective.

Gateways of the first two categories will, in general, require the development of tailor-made software on a per system basis, and as a consequence will be less reliable than regular MAP-links.

6 COSTS

Costs always form an important decision criterion when it comes to a final selection. This section highlights cost aspects of using MAP-based products by presenting the results of two cost evaluations done within TNO.

For an envisaged pilot project within TNO cost has been evaluated for the use of MAP based products and the use of Ethernet-based products. The results of this evaluation were as follows:

<table>
<thead>
<tr>
<th>Category</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Higher</td>
<td>hardware</td>
</tr>
<tr>
<td>Similar</td>
<td>specification, design</td>
</tr>
<tr>
<td>Lower</td>
<td>coding, testing, integration, maintenance, extensions</td>
</tr>
</tbody>
</table>

From this comparison it is expected that the overall costs for MAP based solutions will be less expensive than integrating standard control systems in a multi-vendor environment.

Another evaluation has been performed for an existing flexible manufacturing system. The additional costs of using MAP instead of the existing RS232 links would have required an additional investment on hardware of approximately Dfl. 500.000. Migrating the existing system according to the fine grain solution is expected to cost between Dfl. 1.5M - 2.5M. The fine grain solution implies the replacement of regular terminals by PCs containing MAP interfaces and PC-based MAP gateways for the manufacturing equipment.

The conclusion is that from a cost point of view it is definitely worthwhile considering the use of MAP 3.0.
7 HOW TO START WITH MAP

MAP is undoubtedly a candidate solution to problems in computer integrated manufacturing, especially in a multi-vendor environment. In general we advise the following phases in the introduction of MAP in a production environment:

- strategy study
- a pilot project, including
  - specification
  - implementation
  - education
- regular project.

The first phase consists of a strategy study, where the advantages and disadvantages of MAP in relation to other alternatives are evaluated. This may also include the evaluation of possible network configurations. If the outcome of the strategy study results in the recommendation of MAP, a successive pilot project is recommended, including a specification and implementation phase. If the functionality provided by the MAP solution is different from the existing control system functionality, the pilot project should also be used to familiarize operator staff with this new functionality. After the completion and evaluation of the pilot project, a regular project can be started with good confidence in its successful completion.

8 TNO’S INVOLVEMENT IN MAP

At this moment TNO is, or has been involved in a number of initiatives for MAP pilot projects:

- Cooperation with Philips Center for Manufacturing Technology (CFT center) where TNO staff worked together with Philips staff on the evaluation of the interchangeability of MAP 2.1 products.
- Evaluating the possible use of MAP 3.0 in an existing Flexible Manufacturing System.
- Undertaking a project where MAP 3.0 is used for a small scale robot for the construction of buildings.
- Participation in the MAP working group of Holland Elektronika (HE), which is the division for Electronics and Industrial Automation of the Netherlands Association of the Mechanical and Electrical Engineering Industries FME. This working group is examining the possibility of a joint pilot project to be undertaken by a number of Dutch (based) companies.
- Investigating together with the TU-Delft the possibility of setting up a MAP pilot project.
As may appear from these examples TNO is convinced that MAP 3.0 offers additional possibilities for computer integrated manufacturing and we are glad to see that this insight is shared by a number of other companies and institutes.

9 CONCLUSIONS

MAP 3.0 offers a number of advantages for computer integrated manufacturing applications in a multi-vendor environment. Advantages especially related to reliability and safety are the following:

- uniform functionality for a wide range of computer equipment,
- shortening of the integration and testing phase,
- conformance testing, providing independent certification of MAP 3.0 products,
- relative ease in maintaining and modifying existing systems.

Cost evaluations show that also from this point of view MAP 3.0 is a serious alternative to other available solutions to the communications problem in CIM.

Based on the "walk before you run" principle, a phased approach for the introduction of MAP is recommended.

References

In 1980, the formation of the Safety Science Group within the Department of General Sciences made it possible to develop and apply system safety techniques to research projects in the area of electrical engineering. The Electrical Engineering and Safety Group was founded as an organisation in 1981 in order to stimulate research and education regarding the safety aspects of the design, development and use of electrical or information (sub-)system. To date, the group has generated a number of master's level projects for students, mainly in the area of traffic system safety (road, rail and vessel traffic systems) as well as health care (hospital) systems. These projects are aimed at either system modelling and the derivation of safety assurance requirements or to systematic hazard analysis using an extensive tool kit such as MORT: Management Oversight and Risk Tree. The organisation has stimulated several research projects and supported existing and post-graduate courses where safety is of relevance.

**Functional Soft- and Hardware Safety**

The organisation's major project is aimed at the problems arising when information sub-systems are applied as parts of larger systems. The growing vital role of such sub-systems introduces new types of potentially major hazards and new dimensions of quality assurance in relation to data integrity. The overall system's performance can be violated by insufficient software and hardware reliability. In addition, more attention must be paid to man-machine interfaces in the light of the growing role of people as decision-makers in complex work processes and the dynamics of 'human error'. This project is also connected with the IEC Advisory Commission on Safety (ACOS), Working Group on Programmable Electronic Systems. The Electrical Engineering & Safety Group actively participates in the European Workshop on Industrial Computer Systems, Technical Committee 7 on reliability, safety and security (EWICS TC7).