Control and Diagnosis in Integrated Product Development – Observations during the Development of an AGV

R Stetter¹, A Simundsson²

¹ Hochschule Ravensburg-Weingarten, Weingarten, Germany
² University of Manitoba, Winnipeg, Manitoba, Canada

E-mail: stetter@hs-weingarten.de

Abstract. This paper is concerned with the integration of control and diagnosis functionalities into the development of complete systems which include mechanical, electrical and electronic subsystems. For the development of such systems the strategies, methods and tools of integrated product development have attracted significant attention during the last decades. Today, it is generally observed that product development processes of complex systems can only be successful if the activities in the different domains are well connected and synchronised and if an ongoing communication is present – an ongoing communication spanning the technical domains and also including functions such as production planning, marketing/distribution, quality assurance, service and project planning. Obviously, numerous approaches to tackle this challenge are present in scientific literature and in industrial practice, as well. Today, the functionality and safety of most products is to a large degree dependent on control and diagnosis functionalities. Still, there is comparatively little research concentrating on the integration of the development of these functionalities into the overall product development processes. The main source of insight of the presented research is the product development process of an Automated Guided Vehicle (AGV) which is intended to be used on rough terrain. The paper starts with a background describing Integrated Product Development. The second section deals with the product development of the sample product. The third part summarizes some insights and formulates first hypotheses concerning control and diagnosis in Integrated Product Development.

1. Integrated Product Development

In recent years, powerful methods and algorithms for the control, monitoring and diagnosis of industrial systems and processes have been developed and the potential resulting from the application could be proven. However, as a consequence of the multitude and variety of these methods and tools it is difficult for product developers of future systems and processes to choose the right combination of methods and tools for their specific application. The challenging tasks are to achieve a technical and economical optimum, including a sensible use of available resources and considerations of robustness, life-cycle and sustainability issues. This paper is based on the hypothesis that the well-known methodology integrated product development (IPD) could assist product developers and control or diagnosis engineers in their endeavor.

First the terms control and diagnosis will be explained before the core of IPD will be elucidated. The term control names activities intended to manage, command, direct or regulate the behavior of
devices or systems. Control has been the core of extensive research for many decades. In recent years the techniques of predictive control have found rising attention (compare e.g. Camacho and Bordons [1] or Wang and Boyd [2]). Predictive control usually relies on dynamic models of the process, most often linear empirical models obtained by system identification. In the area of pump systems predictive control can pursue three different objectives: smoothing changes of system states, better coordination of multiple pumps and evaluating decision alternatives.

Diagnosis is usually understood as the process of estimating the object’s condition. Diagnosis is carried out by the estimation of important parameters, by the detection and isolation of faults, by the identification of faults and by the determination what should be done in the case that faults occur. Over the last three decades, the growing demand for safety, reliability, and maintainability in technical systems has drawn significant research in the field of diagnosis. Such efforts have led to the development of many techniques; see for example the most recent survey works (Blanke et al.[3], Isermann [4], Witczak [5], Korbicz et al.[6]). For fault compensation, fault tolerant control methods are proposed which can generally be classified into two types, i.e. Passive Fault Tolerant Control Scheme (PFTCS) and Active Fault Tolerant Control Scheme (AFTCS) (Blanke et al. [3]).

According to Ehrlenspiel [7] integrated product development is a “holistic approach to overcome the problems that arise in product development due to the division of manpower” (Figure 1).

![Figure 1. Integrated Product Development (Ehrlenspiel).](image)

He states that integrated product development is an approach that includes different methods of problem solving, organizational methods of optimizing interpersonal processes and technical methods for the direct improvements of products. Traditional “over the wall” approaches are sequential by nature as illustrated in Figure 1.
The major disadvantage of this approach is that products are developed with a limited exchange of information and ideas, and people late in the sequence do not have any input to earlier decision stages. As a result, poor decisions are made, which resulted in longer "Time to Market" and higher development costs [8]. Modern parallel approaches have developed from simultaneous engineering over concurrent engineering to integrated product development. Andreasen and Hein [9] describe the main activities in the different streams market, product and (production) process, see Figure 2.

Figure 2. Integrated Product Development (Andreasen and Hein).

Integrated product development emphasizes the earliest possible involvement of all "downstream" departments into the design decisions. Now, while "upstream" work processes have to invest more time into planning, "downstream" work processes save time and costs. Several studies have demonstrated that this approach can result in a shorter “Time to Market”. Furthermore an enhanced product quality can be achieved.

Integrated product development is a holistic approach aiming at the creation of complex innovative systems. Several aspects are important part of this approach; Figure 3 gives an overview.

It is important to note that the implementation of the complex approach integrated product development (IPD) cannot just be the simple use of some methods and tools. A strategic implementation of IPD is necessary; similar to concepts such as Total Quality Management (TQM). IPD requires a stringent process orientation; the implementation requires constant top management level support. Two aspects of IPD enable breakthrough success: an open multi-domain communication in connection with a shared multi-domain project responsibility. The backbone of IPD is the application of state of the art processes, methods and tools on all levels of a product development organization. One core element of IPD is the conscious and systematic use of several abstraction levels of product models, which allows the product development engineers to analyse and understand their product from several viewpoints. This also fosters a deep understanding of the essential requirements and functions of the product under development. IPD greatly emphasizes the fact that first solution ideas are in most cases not the optimum solutions. Therefore, the creation of alternative solutions on all abstraction levels is an integral element of IPD. One of the challenges in development processes of complex systems is to ensure an overview of all activities, decisions and intermediate results. IPD proposes a planning on the macro level based on phases with pre-determined results. This approach allows realistic planning and ensures the traceability or the product development progress, but also prevents inefficient planning. Cornerstones of this approach are multi-domain synchronization points, which link the different domains involved in the product development (such as mechanical engineering, control engineering, electrical engineering as well as computer soft- and hardware). IPD also emphasizes the connection of the product models of the different domains and incorporates approaches towards multi-domain product models. Such models also allow multi-domain simulation, which is a main prerequisite for domain-spanning optimum solutions. Many product examples which are now outdated such as tape recorders show that the main causes for failure lie in the product
concept. IPD therefore promotes a special emphasis on abstract levels (requirements, functions, physical effects) of the product models. Last but not least, the consideration of profound changes over time also aims at preventing a concentration on solutions or technologies which can be outdated. Very often wrong hypotheses such as the size of certain market segments (e.g. computer) have led product management to wrong strategic decisions.

All the listed aspects of IPD enable more conscious and more successful product development. However, an emphasis towards control and diagnosis was not yet a central aspect of IPD (consequently it is not yet shown in Figure 3 as an important aspect – it may belong there). In order to improve the integration of these functionalities a long-term development project at the Hochschule Ravensburg-Weingarten – the development of an AGV – was observed.

2. Product Development of an Automated Guided Vehicle AGV
This section describes the product development of an AGV for outdoor used. This development process was observed and leads to some hypotheses presented in Section 3. The AGV makes use of a unique steering system based on torque differences and is suited for rough terrain because of four independent legs with wheels (Figure 4).
2.1. Torque steering system

Autonomous vehicles such as mobile robots and the respective steering systems have been successfully developed and built for some years (Anderson&Jones [11], Ashmore&Barnes [12], Dillard [13]). The distinctive quality of the steering system used is the dynamic behavior. This innovative steering system that is already registered as a patent ([14], [15]) is based on the concept to use the torque of drive motors (more exactly the torque differences between wheels) to steer four independent axles of a vehicle. The principal steering system is shown in Figure 5.

In this example a vehicle consists of four drive motors which are fastened on arms that may freely rotate. These arms have no drive or brake, only an angle encoder is attached at the end of each axle. These angle encoders measure the angle of the motor and the wheel with regard to the vehicle platform. The distinct characteristic of the innovative drive system is the absence of dedicated steering motors. By means of angle encoders applied at the four steering axles and highly dynamic control algorithms it is possible to steer such vehicles only by means of the four drive motors (compare Figure 4).

Each of the wheels on the short axle can be directed into the desired position by means of the torque applied on the wheel. This could take place sequentially for each individual wheel but also simultaneously, if the control allows different torque on all wheels. This characteristic allows simpler and simultaneously more robust vehicle concepts. It is also a main advantage of this concept that the resulting vehicle is able to drive directly in any direction without time and space consuming turning maneuvers.
Furthermore, a vehicle based on the dynamic drive system is able to turn around its own center. This characteristic is very important if cameras or other equipment are mounted on such vehicles which can only be used in a certain orientation. The innovative steering system shares these advantages with Omni drive systems (Ashmore&Barnes [12]), but has reduced friction as well as easier controllability and offers the possibility to determine an exact position and orientation from an analysis of the angles of the steering axes and the angles of the drive wheels (odometry).

2.2. Realization of the Suspension System

The semi-autonomous vehicle is controlled by a small PC-ALIX that it is connected to 4 EPOS motor controllers, one per wheel. For the communication between the controllers it uses the CANopen protocol. It has also two encoders connected which read the angle of the directions from the rear and the front wheels. The front wheels are moved in parallel, due to the fact that they are connected by a pulley. The rear wheels are connected in the same way as the front wheels.

The vehicle moves on 4 wheels. The wheels are located in each corner, as it can be seen in Figure 6.
The diameter of the wheel has been chosen bigger to have a trim and steady movement in different terrains. For the better traction the surface of the wheel is made out of rubber. In Figure 7 it can be seen that the wheels are joined to the vehicle by a traction system made of two arms and one suspension.

This system permits the vehicle to absorb the different heights of the terrain and the impacts against the ground. There are two belt driven pulleys to transmit the movement from the motor to the wheel. This transmission is a reduction system, because the torque required by the vehicle is more than the torque produced by the motor. The front wheels are joined in pairs by a belt pulley system as well as the rear wheels in the same manner. The pair of wheels has the same direction at every moment. When the wheels are aligned in a straight line, they can turn almost 90 degrees in each direction. The belt pulley system can be seen in Figure 7.

The vehicle has two brakes: one on the front wheel pair and the other one on the rear wheel pair which are intended to hold the steering angle when the direction is not changed. This leads to improved dynamic behavior.

2.3. Control System
So-called EPOS controllers are used for the direct control of the drive motors; there is one EPOS per motor. An EPOS is a small-sized, fully digital motion controller [18]. The EPOS 24/5 can control brushed DC motors with digital encoders as well as brushless EC motors with digital Hall sensors and
encoders. An EPOS is specially designed to be a slave in a CANopen network and it can also be operated through a RS-232 communication port. The vehicle has two other additional encoders, to know the angle between the pair of wheels and the vehicle; they are installed on the vertical axis of one wheel from each pair of wheels- one at the front and one at the rear. The master of the vehicle is a small PC-ALIX by Acer Company. It has five USB ports, HDMI, built-in WiFi and a VGA output. It has a hard disk, and a compact flash memory of 4 GB. The vehicle has two batteries of 12V each. They are connected in series giving 24V. They are used to give the power supply to the EPOS and the Encoders. As the motors are connected to the EPOS, the motors rely on the two batteries indirectly. The lithium ion battery pack is used to supply the PC-ALIX. The vehicle has two switches to control the electrical circuit and has a fuse box to provide safety to the electrical installation. An electrical block diagram can be seen in Figure 8.

**Figure 8.** Electric block diagram.
2.4. Control and Diagnosis of the vehicle

The control of the vehicle happens on different levels. A central computer receives a movement wish either directly from the user (manual driving) or improved by a superordinate intelligence (semi-autonomous driving mode) or only from a superordinate intelligence (autonomous driving mode). Based on a mathematical model of the AGV this movement wish is than translated into a speed command to the drive motors as well as desired angles for the wheels (one angle for both front wheels and one angle for both back wheels).

One critical aspect of the control of the AGV was the control of the drive motors based on a proportional-integral-derivative (PID) controller, which is a control loop feedback mechanism which adjusts processes and control outputs through error monitoring and minimization [16]. Tuning a control loop is defined as the adjustment of the PID gains to values that provide the quickest response with a satisfactory amount of oscillation for the system, therefore providing stability to the system. However, tuning is a rather difficult process and various methods can be used. It should also be noted that processes that have a certain degree of non-linearity may benefit from gain scheduling; that is, if the system contains parameters that work well at full-load conditions but have a subpar performance at start-up or no-load, this can be corrected by using different parameters in different operating regions (i.e. by gain scheduling).

One method of tuning is manual tuning. This involves setting the values of the integral ($K_i$) and derivative ($K_d$) controllers to 0, and increasing the proportional value ($K_p$) of the controller until the output of the loop oscillates. $K_p$ should be then set to approximately half that value. Next, increase $K_i$ until any offset is corrected in an acceptable time frame for the process. Finally, increase $K_d$ (if necessary) until the loop reaches the desired output value in an acceptable time frame. Note that most systems accept a slight amount of overshoot in order to reach the desired set-point more quickly, but only in cases in which some overshoot is not an issue. If the system cannot handle any degree of overshoot, an over-damped system is required, which will have a $K_p$ of significantly less than half the $K_p$ that would be used to cause a slight overshoot in the system. The controller used in this project also has a position regulation with a feed forward feature, through which the effectiveness of the PID controller can be improved even further. The variables used in the feed forward mechanism used are the “Velocity Feedforward Factor” ($K_v$) and the “Acceleration Feedforward Factor” ($K_a$). $K_a$ is calculated based on the total inertia of the drive system ($J_{tot} = J_{motor} + J_{load}$) and the motor torque constant ($k_m$). The motor torque constant can be found in the motor catalogue. For the motor used in this project, it is 16,9 mNm/A. The load inertia must be measured in order to calculate the total inertia, and the motor inertia can be taken as the rotor inertia, also given in the motor catalogue.

The Velocity Feedforward Factor is calculated based on a linear load current increases if the angular velocity is enhance by a $\Delta \omega$. This can easily be determined by measuring the current at two different speeds, or assuming the current is zero when the angular velocity is zero. The constants are calculated based on formulae 1, 2, and 3:

$$K_v = \frac{J_{tot} \cdot 1000}{k_m} \left[ \frac{A}{rad/s^2} \cdot 10^{-7} \right]$$  \hspace{1cm} (1)

$$J_{tot} = J_{motor} + J_{load} \left[ gcm^2 \right]$$  \hspace{1cm} (2)

$$K_w = \frac{30 \ \Delta I}{\pi \ \Delta \omega} \left[ \frac{\mu A}{rad/s} \right]$$  \hspace{1cm} (3)

Once calculated, these values are written to the objects “Acceleration Feed-forward Factor”, and the “Velocity Feedforward Factor” in the object dictionary as per instructions given in the Epos Application Note. Once these values are written to objects, the system is now ready for tuning with the tuning wizard available in Epos Studio. This approach led to a satisfactory control behavior.
3. Observations in the project
The development of the AGV took more than one year and several different teams which members from mechanical engineering, electrical engineering and computer science were involved. In this section some observation of this process are reported.

The development processes which could be observed were much more evolutionary than revolutionary. This distinction was suggested by Stetter et al. [17] for mechatronic development processes. In their research they were observing evolutionary processes in different types of industrial development processes. Evolutionary processes are characterized by gradual change and development. On the contrary, revolutionary design can be understood as a sudden or complete change in something. The observed project showed several iterations and many jumps from abstract to concrete levels of product development, e. g. between task clarification and detail design. The success of the project indicates that the central advantages of an evolutionary process – flexibility – did outweigh the risk of an early focus on maybe suboptimal solution.

From later research Stetter et al. [18] also report the importance of fast, early prototyping. Fast, early prototyping requires conscious jumps at early stages of a product development process to concrete prototypes in the sense of physical mock-ups, tests but also software prototypes. In system engineering preliminary studies can describe similar activities. Throughout the project, the question: “How should I develop a control system, if I don’t know the mechanical arrangement?” could be heard frequently from the persons responsible for control and diagnosis functionalities. The collaboration was usually much easier if some physical system was available, even if it only was a rather basic prototype.

A large difference between the documentation needs could be observed. Mechanical engineers needed to provide accurate technical drawings in order to have the parts to be made in the university workshop or by outside companies. Therefore, they were forced to document and rethink their solution before they could touch or test it. Also, because of the production expenses these students tried rather hard to avoid mistakes maybe they also concentrated on “safe” solutions. On the contrary, the students in charge of computer programs initially documented nothing (later on they added some commentaries in the code) and could immediately test and improve the resulting code. Probably as a consequence, they were more open to try new and unexplored ideas.

In all disciplines some forms of trouble shooting could be observed. In the mechanical engineering this was mainly during the assembly process, which was not documented up front. In electrical engineering the cable routing exhibited several instances of trouble shooting. As mentioned before, a large share of the work of the computer scientist was aimed in this direction.

One important prerequisite for successful interdisciplinary work is trust – trust that the other disciplines will stick to the negotiated interfaces and will quickly advise as to any necessary changes. The importance of trust was also reported by Pulm&Pulm et al. [19], [20], which they understand to be a willingness to rely on another party in the confidence that this reliance will provide a positive outcome. They conclude that the main function of trust in product development is achieving a shared understanding and therefore an enhanced effectiveness and efficiency. The development process of the AGV showed usually a certain time span until specialists of different disciplines were able to trust each other. This trust was enhanced, when mechanical engineers were able to show physical models and computer scientists could demonstrate successful programs.

Stetter&Pulm [21] also report that the synchronization of the different disciplines is consequently a major challenge for project management of mechatronic products. The consequences of this challenge could also be frequently observed during the development process of the AGV. Very often the needs for information as well as the intermediate results that were achieved were unclear for students from other scientific disciplines and common project planning was difficult and time consuming. In general Gantt chart planning was used and was in general successful. However, it was frequently unclear which level of detail is appropriate for this kind of planning. Also the controlling of mile-stones was rather difficult as the deliverables at mile-stones were usually not defined or only very roughly defined.
The main problem concerning the **abstract structure** of a mechatronic system is that the mechatronic system has to be structured in different dimensions [21]. The structures have to cover the abstract functional structure, the structure of modules, the mechanical structure (geometry), the electronic structure (“systems”) in the meaning of a network, as well as the software structure (software functions). Ideally, these should be congruent, but in the project there were various reasons why they differed: functions were spread over various components; each structure had to be regarded independently in order to get an optimized system (and together in order to get an optimized product); functions were strongly interconnected and defined from different viewpoints or on different levels. While these problems could in theory be solved by a clear functional structuring, other aspects hindered the success of this approach: the complexity of the mechatronic system did not allow a complete modeling of the system; the involved disciplines had a different thinking; many problems could only be solved by bypass solutions regarding the structure. In the project the abstract structure was never really documented and evolved by trial and error.

### 4. Summary and Outlook

One of the main trends in the evolution of technical systems is a development towards higher automation, i.e. towards a lower amount of required human interaction with the system [22], [23]. This trend requires extremely powerful control algorithms. The control of an autonomous or semi-autonomous system has to be able to compensate for external influences such as disturbances and noise. Furthermore the control of such systems has to be able to detect and identify certain faults and has to be able to compensate for their consequences or to transform the system into a safe state. Therefore, extremely powerful diagnosis capabilities are also required for autonomous or semi-autonomous systems. Conventional product development processes do not emphasize this prominent role of control and diagnosis. Today these capabilities are frequently added to systems in rather late stages of the product development processes, thus leading to sub-optimum results. Consequently, efforts toward a stronger integration of the capabilities control and diagnosis into the product development processes are mandatory.

The main intention of this paper is to describe a product development project including control and diagnosis functionalities – the development of an novel automated guided vehicle with unique properties and unique technical solutions – as well as to report observations during this process and derive first hypotheses about a more efficient and effective integration of the development of control and diagnosis functionalities into product development processes. As these observations and hypotheses are in this stage only based on one product development process they are preliminary. The main intentions of this paper were to underline the paramount importance of a stronger integration of control and diagnosis into product development processes and to present central aspects and challenges in the integration endeavor. Further research is planned in order to test the hypotheses as well as both to widen the scope and to deepen the investigation by means of analyzing product development processes.

### References

[1] Camacho, E., Bordons, C.: Model Predictive Control. Berlin: Springer, 2004.
[2] Wang, Y., Boyd, S.: Fast Model Predictive Control using Online Optimization. In: Proceedings of the 17th World Congress. The International Federation of Automatic Control. Seoul, Korea, July 6-11, 2008.
[3] Blanke, M., Konnaert, M., Lunze, J., and Staroswiecki, M.: Diagnosis and Fault–Tolerant Control. Berlin: Springer, 2006.
[4] Isermann, R.: Fault–Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance. Berlin: Springer 2005.
[5] Witczak, M.: Modelling and Estimation Strategies for Fault Diagnosis of Non–Linear Systems: From Analytical to Soft Computing Approaches. Lecture Notes in Control and Information Sciences. Berlin: Springer 2007.
[6] Korbicz, J., J.M. Kościelny, Z. Kowalczyk and W. Cholewa: Fault Diagnosis: Models, artificial intelligence methods, applications. Springer: Berlin, 2004.
[7] Ehrlenspiel, K.: Integrierte Produktentwicklung - Methoden für Prozessorganisation, Produktentwicklung und Organisation. München: Hanser, 1995.
[8] Schätz, C.: A Methodology for Production Development - The Body of Knowledge Approach. Trondheim: Norwegian University of Science and Technology, 2006.
[9] Andreasen, M. M.; Hein, L.: Integrated Product Development. New York: Springer 1987.
[10] Mandhyani, T.; Kubendra Prasad, P. K.: Development of a Semi-Autonomous outdoor vehicle. Project Report, Hochschule Ravensburg-Weingarten, 2012.
[11] Anderson, K. Jones, C.: “Autonomous Robotic Vehicle Design”, Proceedings of the Southwest Regional ASEE Conference, ASEE 2005.
[12] Ashmore, M., Barnes, N.: “Omnidrive robot motion on curved paths: The fastest path between two points is not a straight line”, Proceedings of the Australian Joint Conference on Artificial Intelligence, Dec., 2002, pp. 225-236.
[13] Dillard, W.: “Revisiting the Autonomous Robot. Finding the Engineer within the Student”, Proceedings of the 2004 AEE Conf., Salt Lake City, 2004.
[14] Patent „Fahrzeug“ Aktenzeichen DE 10 2005 057 635 A1, Deutsches Patent- und Markenamt, München.
[15] Patentanmeldung „Fahrzeug“ Aktenzeichen EP 1 954 516 A1 und WO 2007/062858 A1, Europäisches Patentamt, München und World Intellectual Property Organization, Genf, Schweiz.
[16] Simundsson, A.: Summary of Project Work. Project Report, Hochschule Ravensburg-Weingarten, 2014.
[17] Stetter, R., Möhringer, S., Pulm, U.: A Comparison of Evolutionary and Revolutionary Approaches in Mechatronic Design. In: Proceedings of the 18th International Conference on Engineering Design (ICED 11) Vol. 1, 2011, pp. 221-232.
[18] Stetter, R., Möhringer, S., Günther, J., Pulm, U.: Investigation and Support of Evolutionary Design. Accepted for the 20th International Conference on Engineering Design (ICED 15), 2015.
[19] Pulm, U.; Stetter, R.: Pressure and Trust in Competitive Engineering. In: Norell Bergendahl, M.; Grimheden, M.; Leifer, L.; Skogstad, P.; Lindemann, U. (Eds.): Proceedings of the 17th International Conference on Engineering Design (ICED'09), Vol. 9., 2009, pp 69 – 80.
[20] Pulm, U.; Stetter, R.: Influences of pressure on cooperative product development. In: International Journal of CoCreation in Design and the Arts, 1745-3755, Volume 5, Issue 3, 2009, pp 193 – 210.
[21] Stetter, R.; Pulm, U.: Problems and Chances in Industrial Mechatronic Product Development. In: Norell Bergendahl, M.; Grimheden, M.; Leifer, L.; Skogstad, P.; Lindemann, U. (Eds.): Proceedings of the 17th International Conference on Engineering Design (ICED'09), Vol. 5., 2009, pp 97 – 108.
[22] Herb, R.; Herb, T.: Kohnhauser, V.: TRIZ. miVerlag, 2000.
[23] Diers, S.: Systematik in der Anwendung der Evolutionsprinzipien (TRIZ). Triz-online: 2006.