Development of a Trajectory Kernel for Autonomous Vehicles

Lothar Seybold*, Andrzej Pieczyński**
Andreas Paczynski*, Jaroslaw Krokowicz***, Ralf Stetter*****

** RAFI GmbH & Co. KG, 88276 Berg, Germany (e-mail: lothar.seybold@rafi.de).*
*** Uniwersytet Zielonogórski, 65-417 Zielona Góra, Poland (A.Pieczynski@issi.uz.zgora.pl)*
****Hochschule Ravensburg-Weingarten, 88241 Weingarten, Germany (e-mail: stetter@hs-weingarten.de)*

Abstract: Future production concepts which are currently developed in the scope of "agile production" will require autonomous working and transportation platforms which dispose of much higher flexibility and robustness than current autonomous guided vehicle (AGV). One the one hand, such flexibility and robustness will only be possible if an optimal functionality and interoperability of the monitoring, planning, control and diagnosis systems can be achieved. On the other hand, a rapid multidimensional trajectory planning is inevitable in order to achieve the required flexibility and robustness. In the first part of this paper a hierarchical and distributed concept and realization proposal will be presented aiming at optimized functionality and interoperability. Based on the concept and the realization proposal, a system for multidimensional trajectory planning called "trajectory kernel" in a basic and an enhanced form will be presented in the second part. The paper concludes with an outlook to the control of multiple autonomous vehicles applying Max-plus-Algebra. It is important to note that the focus of this paper is not a description of the respective algorithms but an explanation of the framework and the application. Investigations about the industrial application of intelligent systems for monitoring, control and diagnosis (Kleinmann et al. 2009, Stetter&Kleinmann 2011) have shown that frequently not the lack of the optimum algorithms but the missing integration in existing infrastructure and a missing higher level concept are important causes for the low application ratio of such intelligent systems. Consequently, research aiming at optimized integration and searching for high level concepts is obviously as desirable as the search for new and improved algorithms.

Keyword: Monitoring, Planning, Control, Diagnosis, Autonomous Vehicles

1. INTRODUCTION

This paper describes two components for a control and diagnosis system for autonomous vehicles: a hierarchical and distributed concept for a system architecture and a software component which is generating paths for such vehicles which can be referred to as "trajectory kernel". The trajectory kernel is one of the cornerstones of the proposed realization concept for the hierarchical and distributed system and allows a basic but functional and quick trajectory generation for autonomous vehicles. The main distinctions of this kernel to the numerous other trajectory generation algorithms are a specific strategy for avoiding unnecessary acceleration and deceleration, the eased application of different priorities i. e. optimization criteria (time/energy/production space) and the use of basic shape combination, In the enhanced trajectory kernel additionally a continuous total acceleration strategy is applied in order to achieve less interruptions on the vehicle and on transported goods.

The planning and control of a number of autonomous vehicles is a challenging task because it is characterized by a large number of degrees of freedom, by multi-objective optimization and consequently demanding complexity. A promising approach to deal with this challenge is the application of max-plus algebra to find an optimal solution in particular when it comes to real systems with their distinctive non-linearities.

The intended areas of application are autonomous vehicles in future production systems which follow the paradigm "agile production". Agile production can be defined as the capability of surviving and prospering in a competitive environment of continuous and unpredictable change by reacting quickly and effectively to changing markets, driven by customer-designed products and services (Cho et al. 1996). Agile production requires rapid changeover from the assembly of one product to the assembly of a different product. This needs a rapid hardware changeover by robots, flexible part feeders, modular grippers, and modular assembly hardware. Agile manufacturing needs intelligent sensing and decision making systems capable of automatically performing many tasks traditionally executed by human beings (Gunasekaran 1999).

The ever increasing product variety will is expected to change the production systems completely in the next two decades. Especially in the area of passenger cars, electro-mobility will lead to a larger product variety, because, on the one hand, this technology allows enhanced modularity and thus adaptability and variability, and on the other hand, the limited range of such vehicles which is largely depending on the weight, will lead to solutions closely suited to the different customer needs.

One prominent approach to realize a flexible (flexible with respect to variety) production systems is the application of transportation platforms which, on the one hand, can transport the different components of a product, and, on the other hand, even can move production stations such as manipulators or welders. Figure 1 shows two autonomous vehicles as production platforms: the left one transporting a product component (in this case a car body) the right one a production station (in this case a manipulator).

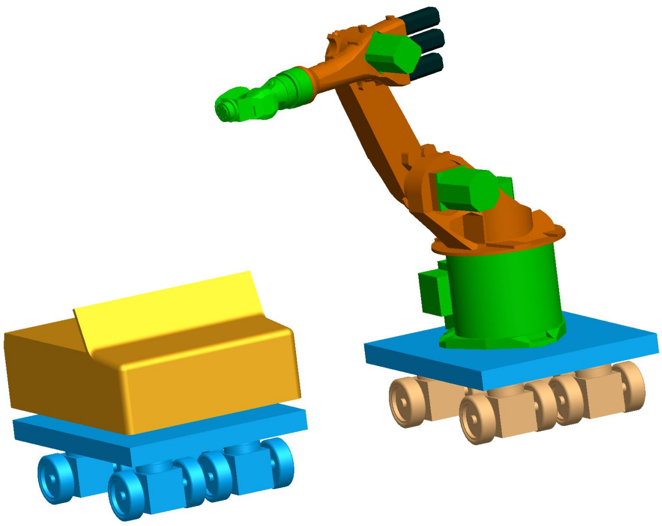

Fig. 1. Autonomous transportation and working platform

2. DEFINITIONS AND STATE OF THE ART

As a basis for the further discussions the terms monitoring planning, control and diagnosis are defined and important sources are listed in this section.

The notion **monitoring** summarizes all kinds of systematic observation, surveillance or recording of an activity or a process by any technical means. In the area of autonomous vehicles monitoring could be understood as a systematic collection of data concerning the state of certain physical characteristics such as distance, speed, acceleration, temperature, vibrations, torque, currents, voltage, power consumption, current gradient, velocity gradient, etc. In leading industries such as computer chip production or car manufacturing today usually nearly all operation data of the productions systems are being monitored for the three main reasons safety, efficiency and planability. Such monitoring of autonomous vehicles is currently only realized to a limited degree but could be an additional function of a control and diagnosis system.

Planning is essentially the process of predefining activities in the future. In the area of autonomous vehicles planning can apply to the assignments of tasks to an autonomous vehicle (e. g. “transport part x from station y to station z”), the planning of paths and trajectories (e. g. “drive along a certain line”) and the planning of the driving behaviour (e. g. “drive with velocity x and acceleration y and maximum acceleration change z”). Obviously these planning activities concern different levels of detail and the required performance in terms of data amount and data speed differs greatly. This fact makes a holistic approach desirable.

The term “**control**” names activities which serve to manage, command, direct or regulate the behaviour of devices or systems and 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&Bordons 2004, Wang&Boyd 2008). Predictive control usually relies on dynamic models of the process, most often linear empirical models obtained by system identification. In the area of autonomous vehicles predictive control can pursue three different objectives:

- smoothing changes of system states,
- better coordination of multiple autonomous vehicles and
- evaluating decision alternatives.

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. 2006, Isermann 2005, Witczak 2007, Zhang and Jiang 2008, Korbicz et al. 2004). For fault compensation in general fault tolerant control methods are proposed which can be classified into two types, i.e. Passive Fault Tolerant Control Scheme (PFTCS) and Active Fault Tolerant Control Scheme (AFTCS) (Blanke et al. 2006, Zhang and Jiang 2008).

3. HIERARCHICAL AND DISTRIBUTED CONCEPT

Discussion with leading customers of autonomous vehicles made clear that control and diagnosis systems will only be adapted in future if they are an integral part of the production system information infrastructure, namely the Enterprise Resource Planning (ERP) system and Manufacturing Execution System (MES). Enterprise resource planning (ERP) is an integrated computer-based system used to manage internal and external resources including tangible assets, financial resources, materials, and human resources. It is a software architecture which purpose is to facilitate the flow of information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders. Built on a centralized database and normally utilizing a common computing platform, ERP systems consolidate all business operations into a uniform and enterprise wide system environment (Bidgoli 2004). In all kinds of companies ERP systems usually present the top level of control within production. On the next level below are Manufacturing Execution Systems (MES). Boldly speaking, an ERP system defines what is to be produced within a given time period and the execution level (MES) takes this planning output and executes this plan on a near real-time/on-line basis (McCellan 1997). This control loop from ERP over MES to the real production operations is usually not closed today. An advanced monitoring would contribute to offer the major advantages usually connected with closed loop control (Ward 2007):

- disturbance compensation,
- guaranteed performance even with model uncertainties, when the model structure does not match perfectly the real process and the model parameters are not exact,
- stabilisation possibilities for unstable processes,
- reduced sensitivity to parameter variations and
- improved reference tracking performance.

In the following section a hierarchical concept is presented which is intended to combine the functionalities of monitoring, planning, control (including predictive control), diagnosis (including predictive diagnosis) into a sensible system structure for a holistic operation of industrial systems including autonomous vehicles.

The insights from numerous discussions with representatives of leading companies indicate that holistic monitoring, planning, control and diagnosis systems for autonomous vehicles in industrial applications need to be integrated in the information system infrastructure of the respective industrial company. In these applications like in most other industry a hierarchical system can be observed. Figure 2 shows a proposal of a sensible hierarchy of the levels of these information systems in form of a pyramid where the higher levels are characterised by higher data quantity and the lower levels by higher data speed.

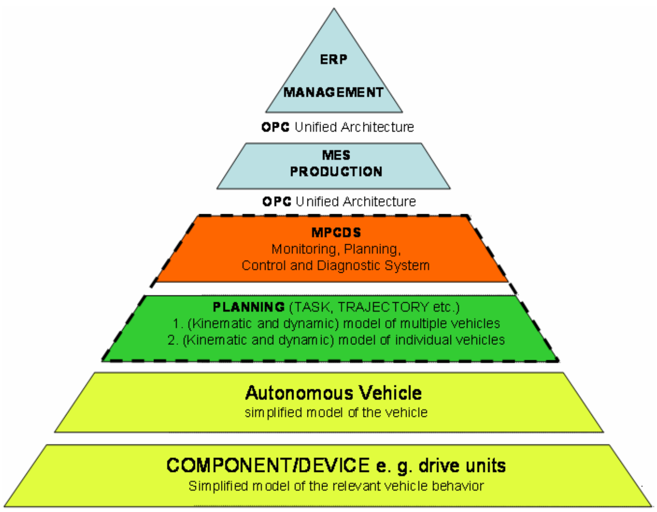

Fig. 2. Distributed and hierarchical concept

On the highest level the Enterprise resource planning (ERP) system can be found. It is not present in all kind of enterprises and is sometimes referred to with different names. The main function is always the same: this level concerns the planning of the entities to be produced on a not time-critical level. On the next lower level the production of these entities is executed by a plant control system which fulfils similar tasks than a Manufacturing Execution System (MES) Sometimes the two highest levels are realized in only one system.

On the next level is the first core of the proposed concept – the monitoring, planning, control and diagnosis system (MPCDS) for a section of a system which usually is including a number of autonomous vehicles. The MPCDS level with the planning level is described in detail in the respective subsection. The next lower level is the single autonomous vehicle. The lowest level contains components of a vehicle with an own local intelligence. Such a concept of distributed intelligence (compare Seybold et al. 2009) is very frequent in the upcoming age of ubiquitous computing and offers many advantages such as flexibility and reliability. On this level the real-time control has to take place and the most important safety functions should be realized on this level for the sake of a quick reaction. However, a number of aspects have to be considered for a sensible structure on this level. These aspects are discussed in the respective subsection.

4. IMPLEMENTATION CONCEPT

The communication at the higher levels of the pyramid is today frequently realized applying OPC (OLE - Object Linking and Embedding - for process control - compare e. g. Iwanitz&Lange 2005). OPC is an open communication standard, which is used in industry automation and in information systems for process management as well as in business administration systems. OPC allows the use of unified access methods and data descriptions for technological processes. At the lower levels of the hierarchical and distributed concept, real-time capabilities are inevitable because of, for instance, the necessary reaction times in dangerous situations such as collisions. Ford et al. (2009) report two approaches for highly reliable, real-time capable systems: Real-time CORBA and DDS. CORBA (Common Object Request Broker Architecture) is a standard of the Object Management Group (OMG) which aims to distribute functionalities over a system (OMG 2011a). The real-time standard CORBA of the OMG includes among others a predictable memory management. DDS (Data Distribution Service) is as well a standard of the OMG (Pardo-Castellote 2003). This approach is based on a data centered, real-time capable “Publish-Subscribe” system architecture (compare Ryll&Ratchev (2011) - Figure 3).

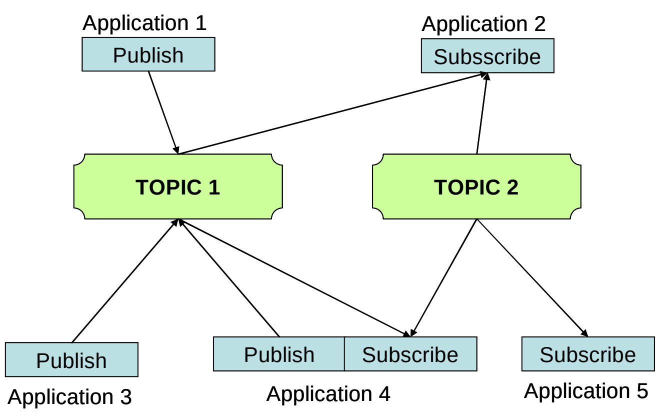

Fig. 3. Publish-suscribe-paradigm

In such environments clients of a system may subscribe to certain information areas (e. g. subscribe to a temperature information of a certain sensor). These information areas are then published by other clients.

Different implementations of the publish-subscribe-paradigm have emerged for supporting the needs of different application domains (Corsaro 2011). The Data Distribution Service for Real-Time Systems (DDS) is a standard for that addresses the needs of mission- and business critical applications, such as, financial trading, air traffic control and management, and complex supervisory and telemetry systems. The OMG DDS standards family is today composed, as shown in Figure 4, by the DDS v1.2 API (OMG 2011b) and the DDS Interoperability Wire Protocol (DDSI v2.1 - OMG 2011c). The DDS API standard guarantees source code portability across different vendor implementations, while the DDSI Standard ensures on the wire interoperability across DDS implementations from different vendors. The DDS API standard defines several different profiles (see Figure 4) that enhance real-time communication with content filtering, persistence, automatic fail-over, and transparent integration into object oriented languages (Corsaro 2011).

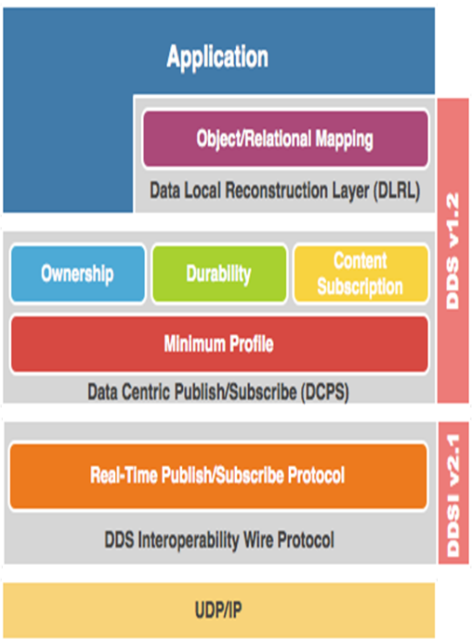

Fig. 4. The DDS-Standard (Corsaro 2011)

Specific advantages of this approach are the modular design, the loose coupling of the clients, the open interfaces and the capability to guarantee specific performance requirements (quality of service). As a consequence, this approach is well suited for realizing the middle (MPCDS) and lower levels of the hierarchical and distributed concept. Figure 5 shows one possibility for the realization of the concept which is based on DDS.

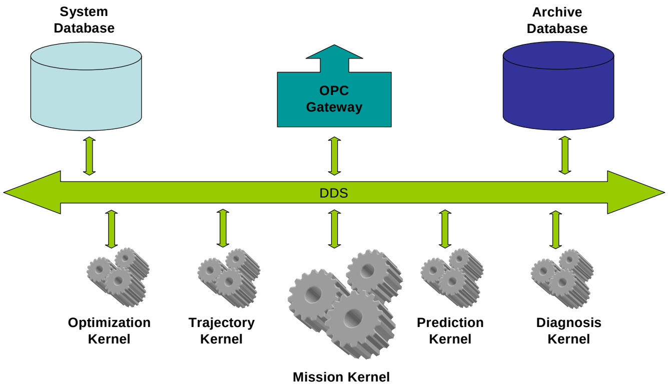

Fig. 5. Implementation concept

The central level of the hierarchical and distributed concept is the system for monitoring, planning, control and diagnosis (MPCDS). In this system several functionalities are integrated which require different algorithms and complex function blocks for different tasks e. g. for the calculation and optimization of trajectories of autonomous vehicles. Additionally, this system needs to store and provide data of different quality and quantity (ranging from configuration information such as the current weight of a vehicle to descriptions of current and planned missions of the vehicle). Already on this level real-time capabilities are required for safety reasons. One possible approach to interlink different components (algorithms and function blocks as so-called "kernels", data bases, actors and subordinate systems as well as an interface to OPC and via this to the MES and the ERP system) presents the standard DDS (Data Distribution Service).

Already today several implementations of DDS could be realized: DDS is, for instance" required by the US Department of Defense (DoD) as standard. The potential for the realization can therefore be considered rather high. This service allows to link different applications and data bases and to achieve real-time capabilities in this endeavor.

On important cornerstone of this concept is the trajectory kernel. A basic and an enhanced approach for the realization of this kernel are described in the following sections.

5. BASIC TRAJECTORY KERNEL

The trajectory kernel is a function block (a software component) which can be used for the realization of several functionalities. For planning it allows to calculate the minimum time to realize a certain transportation task. For control purposes it delivers the main basis for steering the vehicle. It can also be used for monitoring and diagnosis if a theoretical trajectory is generated parallel to driving maneuvers and if deviations are detected (compare residual generation). Therefore, this function block may be used many times, therefore a good performance is desirable. Consequently, in this project a concept of a very basic trajectory kernel which still delivers relatively good trajectories was developed.

For autonomous vehicles trajectories need to be generated in a two-dimensional plane. A trajectory can be described as a parametric curve (Biagiotti&Melchiorri 2008)

$$\vec{p} = \vec{p}(u) \quad (1)$$

The trajectory is completely defined only when the motion law is provided

$$u = u(t) \quad (2)$$

Here the function u(t) is a function which derivatives are velocity, acceleration and jerk.

This trajectory kernel generates such parametric curve and the respective function and is based on the use of the simple geometric shapes circle and tangential line (motion primitives – Figure 6).

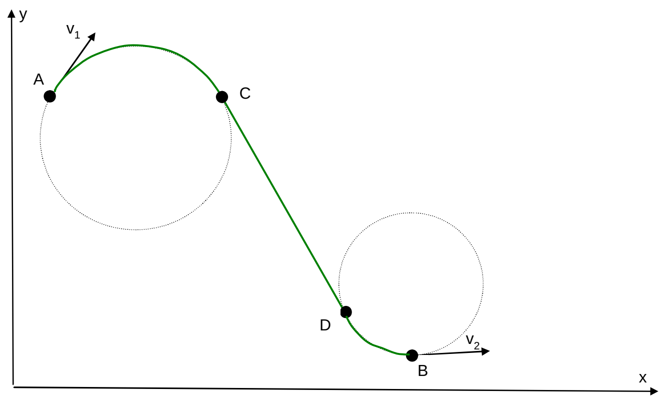

Fig. 6. Basic trajectory kernel – motion primitives

Such simple geometric shapes – also called motion primitives – were applied multiple times in research and industry (compare e. g. the so-called Dubins vehicles (Dubins 1957)). For the basic trajectory kernel additionally some sensible simplifications are made:

- The radius of the basic shape "circle" is always determined by the velocity of the vehicle. The friction contact of a wheel to the floor only allows the transfer of a certain amount of centrifugal forces. In this kernel a given friction parameter is used in order to determine an appropriate circle radius for a certain speed (Figure 5).
- Acceleration and deceleration are only realized on the straight tangential lines. In this way the smallest circles possible can be used, as the friction contact in curves is not additionally loaded by inertia forces resulting from acceleration and deceleration.
- For the acceleration and deceleration on the straight line a trapezoidal trajectory was applied (compare Biagiotti&Melchiorri 2008). This profile is a very common method to obtain trajectories with a continuous velocity profile and is the simplest possibility which can be practically realized (Figure 7). Jumps in the acceleration occur and lead to interruptions on the vehicle and its load. This problem will be addressed in the next section - enhanced trajectory kernel.

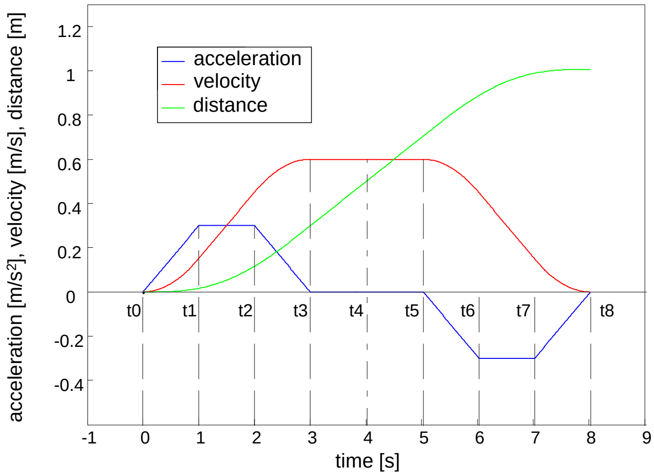

Fig. 7. Basic trajectory kernel – acceleration and deceleration

For the operation of the basic trajectory kernel the initial position and velocity vector and the desired position and velocity vector needs to be given. Usually the initial point will be the current position and the desired position will be the location where some kind of an activity has to be executed. Additionally values such as maximum total acceleration and maximum speed need to be given. Such information has to be processed by means of the mission kernel (compare Fig. 5) from the mission list from the MES. The application of the trajectory kernel can deliver the following results:

- a table of position and velocity vectors with a line showing a given time interval (e. g. 50ms)
- the position of the motion primitives circle and line
- the minimum time needed to perform the mission under the given conditions (maximum velocity and maximum acceleration).

Figure 8 summarizes the main considerations of the simplified trajectory kernel.

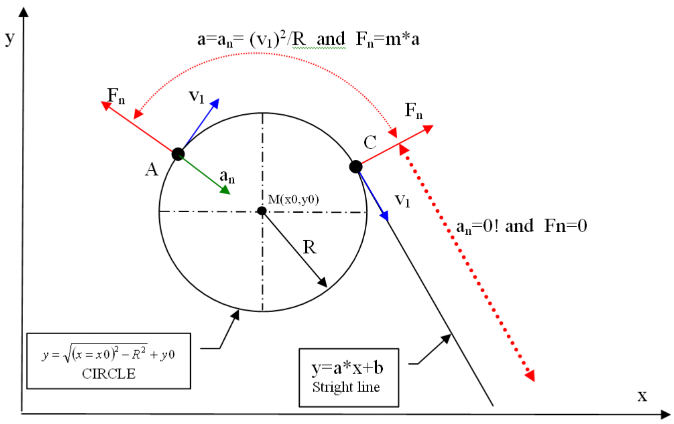

Fig. 8. Basic trajectory kernel – main considerations

The trajectory kernel can then also be used to create alternative trajectories using different leading criteria e. g. time or energy consumption.

Today a large number of algorithms for trajectory synthesis are available – a good overview is given by Biagiotti&Melchiorri (2008). The distinctive quality of the presented basic trajectory kernel in comparison to the numerous other trajectory generation algorithms are the straight-forward strategy for avoiding unnecessary acceleration and deceleration (acceleration and deceleration only on the tangential lines), the possibility to set different priorities i. e. optimization criteria (time/energy/production space) and the use of basic shape combination. These characteristics and strategies which consider the physical properties of autonomous vehicles largely reduce degrees of freedom and allow fast calculations but still deliver a sensible trajectory. Due to the analytical nature of the approach a safe behavior of the vehicle can be guaranteed.

6. ENHANCED TRAJECTORY KERNEL

As stated above, the main problem of the basic trajectory kernel can be the interruptions of the vehicle and its load due to the jumps in the acceleration profile (infinite jerk). In the case of vehicles it is sensible to consider the two components path acceleration (resulting form acceleration and deceleration) and normal acceleration (resulting from the curvature of the path). The effect to the vehicle and its load is the same; it is therefore desirable to have smooth changes of the resulting acceleration vector (continuous total acceleration strategy). Figure 9 shows the approach for the enhanced trajectory kernel.

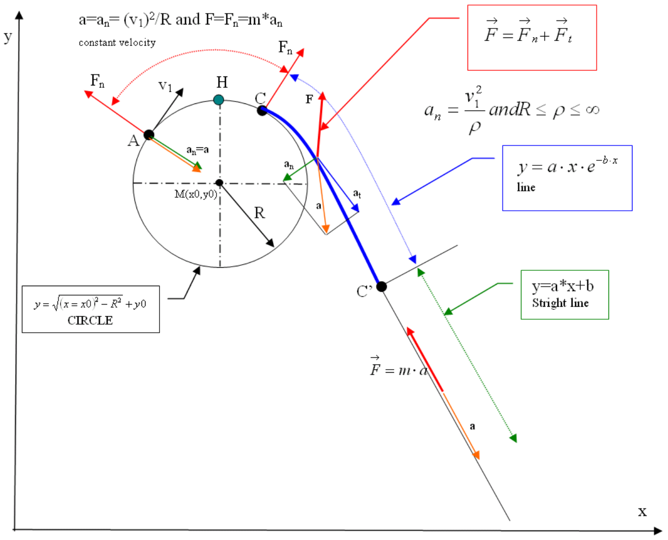

Fig. 9. Enhanced trajectory kernel

Two main points characterize the considerations: firstly it is now necessary also to consider the acceleration or deceleration in the initial and desired position – i. e. to know the rotation of the velocity vector at the beginning and the end. Secondly at all instances of the trajectory where a change from a circle to a line of vice versa is necessary a smoothing sections has to be integrated. Elaborate investigations have show that an e-function is an appropriate solution to realize such smoothing sections (Figure 9). Another possible approach would be, for instance, the clotoid function, which was applied for rail tracks for the sake of smoothing between curves and straights. The clotoid function is the curve defined by the parametric equations (Jabuka 2003):

$$x(t) = a\sqrt{\pi}\int_0^t \cos\frac{s^2\pi}{2}ds \quad (3)$$

$$y(t) = a\sqrt{\pi}\int_0^t \sin\frac{s^2\pi}{2}ds \quad (4)$$

However, these functions are very complicated (Jabuka 2003) and require to solve an integral and were therefore not chosen for a quick trajectory kernel.

To conclude, the continuous total acceleration strategy of the enhanced trajectory kernel applied in order to achieve fewer interruptions on the vehicle and on transported goods. The changes of the trajectory are minor and can be applied for instance only during operation but can be ignored for rough mission planning, for collisions forecast and preliminary avoidance and other activities in the preparation a single or multiple missions of autonomous vehicles.

7. APPLICATION OF MAX-PLUS-ALGEBRA

Within the trajectory kernel numerous assessments and decisions are required to provide a continuous, jerk and collision free trajectory. Above that, the trajectories have to follow different priorities i. e. optimization criteria (time/energy/production space) and apply the appropriate basic shape combination. Furthermore, the trajectory kernel needs to overcome unforeseen events based on the principles mentioned above.

To allow such a flexible and goal oriented optimization of the trajectories, a decision and scheduling system has to be introduced. With the intention of having a fast calculation strategy only a certain part of this system can be part of the trajectory kernel. The second part will be an element of the optimization kernel (compare Fig. 5).

A promising approach to deal with these challenges is the application of the max-plus algebra, which is already successfully deployed in scheduling linear and cyclic processes.

The max-plus algebra or better dioid algebra proves to be one of the best suited tools to treat nonlinear or non-smooth synchronization or optimization problems. Furthermore it also proofs to be suitable for the analysis of deviations. The max-plus-algebra and its application to polynomial and rational functions are mainly based on the work of Cuninghame-Green (1979) and Baccelli et al. (2001). The general applicability for the control of autonomous vehicles was shown by Zaremba et al. (1997), In a first step the focus is on the analysis and optimization of the trajectory represented by a basic 6-parameter dynamic kinematic model of the vehicle, with the state vector $x(t_k)$, including position x, y orientation Θ, and their time derivatives, together with the transition matrix $\Phi(t_k)$.

$$x = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ \Theta \\ \dot{\Theta} \end{bmatrix} \quad \Phi = \begin{bmatrix} 1 & \Delta t & & & & \\ & 1 & & & & \\ & & 1 & \Delta t & & \\ & & & 1 & & \\ & & & & 1 & \Delta t \\ & & & & & 1 \end{bmatrix}$$

Further steps will extend the assessment and optimization of the enhanced trajectory represented in Fig. 9. This step will also involve the min-max analysis of the discrete landmarks introduced in trajectory planning, as well as the min-max analysis of the dependencies of different trajectories.

A subset of this analysis will end up in a pointwise maximum (minimum) of trajectories corresponding to Baccelli (2001).

Based on a rationally closed set, with $h \in K(\alpha)$, there exits $n \in N$, B, $C \in K^n$ and $A_i \in K^{nxn}$, , i=1, …., l, such that:

$$h = C' \left(\bigoplus_{i=1}^{l} \alpha_i A_i \right)^* B \quad (5)$$

The next step will target on the scheduling of vehicles represented by their properties. Different priorities are supposed to create different maxima in the analysis of the transfer functions.

8. SUMMARY

Agile production is leading to new challenges for the trajectory planning. In future production concepts autonomous vehicles with unfinished and finished products and their components and other autonomous vehicles with manipulators and tools may move freely in two-dimensional spaces. This requires new paradigms of planning, optimising and negotiating trajectories for these vehicles.

The presented work is based a hierarchical and distributed concept and realization proposal aiming at optimized functionality and interoperability. A core element of the presented work is a multi-use system for multidimensional trajectory planning called “trajectory kernel” in a basic and an enhanced form. Innovative computation strategies based on different characteristics of the max-plus-algebra may allow to the control multiple autonomous vehicles and optimise their trajectories.

REFERENCES

Baccelli, F. L., Cohen G., Olsder G. J., Quadrat J. P.: Synchronization and Linearity: An Algebra for Discrete Event Systems, John Wiley and Sons, 2001.

Biagiotti, L., Melchiorri, C.: Trajectory Planning for Automatic Machines and Robots. Berlin: Springer, 2008.

Bidgoli, H.: The Internet Encyclopedia, Volume 1, New York: John Wiley & Sons, 2004.

Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M.: Diagnosis and Fault–Tolerant Control. Berlin: Springer, 2006.

Camacho, E., Bordons, C.: Model Predictive Control. Berlin: Springer, 2004.

Cho, H., Jung, M., Kim, M.: Enabling technologies of agile manufacturing and its related activities in Korea, Computers and Industrial Engineering 30 (3) (1996) 323-334.

Corsaro, A.: The Data Distribution Service for Real-Time Systems. White Paper, Access 2011.

Cuninghame-Green R. A.: Minimax Algebra. Berlin: Springer, 1979.

Dabrowska, A., Kleinmann, S.: Analysis of Possible Applications of the Advanced Modelling and Diagnosis System AMandD in German Industry. In: Proceedings of the 7th Workshop on Advanced Control and Diagnosis, ACD’2009, 19. und 20. November 2009, Zielona Góra, Poland.

Diwakar, S. Essawy, M.A. Sabatto, S.Z.: An intelligent system for integrated predictive diagnosis. In: Proceedings of the Thirtieth Southeastern Symposium on System Theory, 1998.

Dubins, L. E.: On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. In: America Journal of Mathematics, Vol. 79, pp 497-516, 1957.

Ford, B.; Bull, P.; Grigg, A.; Guan, L.; Phillips, I.: Adaptive Architectures for Future Highly Dependable, Real-Time Systems. 7th Annual Conference on Systems Engineering Research 2009 (CSER 2009).

Greenfield, A: Everyware - The dawning age of ubiquitous computing. Peachpit Press, 2006.

Gunasekaran, A.: Agile manufacturing: A framework for research and development. In: International Journal of Production Economics 62 (1999) 87-105.

Isermann, R.: Mechatronic Systems: Fundamentals. Berlin: Springer, 2005.

Isermann, R.: Fault–Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance. Berlin: Springer 2005.

Iwanitz, F., Lange, J.: OPC: Grundlagen, Implementierung und Anwendung. Heidelberg: Hüthig, 2005.

Jabuka, S.: Classical Examples of Parametric and Polar Curves. New York: Columbia University, 2003.

Kleinmann, S.; Dabrowska, A.; Koller-Hodac, A.; Leonardo, D.: Model of a combined pump and drive system for advanced control and diagnosis. In: Proceedings “Mechatronic Systems and Materials” (MSM 2010).

Kleinmann, S., Dabrowska, A., Hoffmann, M., Kühn, H., Koller-Hodac, A., Stetter, R.: Advanced Control of Industrial Pump Systems. In: Proceedings of the 7th Workshop on Advanced Control and Diagnosis, ACD’2009, 19. und 20. November 2009, Zielona Góra, Poland.

Korbicz, J., J.M. Kościelny, Z. Kowalczuk and W. Cholewa: Fault Diagnosis: Models, artificial intelligence methods, applications. Springer: Berlin, 2004.

Koscielny J..M., Syfert M., Wnuk P.: Advanced monitoring and diagnosis system "AmandD", In: Proceedings of SafeProcess, Beijing, 2006.
McCellan, M.: Applying Manufacturing Execution Systems. Boca Raton: CRC Press, 1997.
Object Management Group, Inc.: CORBA Basics (Access 2011 - 2011a).
Object Management Group, Inc.: Data Distribution Service for Real-Time Systems Specification. DDS v1.2 (Access 2011 - 2011b).
Object Management Group, Inc.: Data Distribution Ser vice Interoperability Wire-Protocol Specification. DDSI v2.1 (Access 2011 - 2011c).
Pardo-Castellote, G.: OMG data-distribution service: Architectural overview. In ICDCSW '03: Proceedings of the 23rd International Conference on Distributed Computing Systems, Washington, DC, USA, 2003. IEEE Computer Society.
Seybold, L., Krokowicz, J., Pieczyński, A., Paczynski, A, Stetter R.: Prinzipien für das Monitoring, die Planung, Regelung und Diagnose von fahrerlosen Transportsystemen. In: Automation 2011. VDI-Berichte 2143. Düsseldorf: VDI, 2011 (S. 137 – 140).
Seybold, L., Pieczyński, A., Paczynski, A, Stetter R.: Concept of an advanced monitoring, planning, control and diagnosis system for autonomous vehicles. In: Simani, S.; Bonfè, M.; Castaldi, P.; Mimmo, N. (Eds.): Proceedings of the 8th Workshop on Advanced Control and Diagnosis, ACD'2010, 18. and 19. November 2010, Ferrara, Italien (S. 107 – 112).
Seybold, L., Krokowicz, J., Stetter R.: Advanced Control and Diagnosis for Mobile Robots. In: Kluger, K.; Mache, E.; Pawliczek, R. (Eds.): Proceedings of the 6th Conference on Mechatronic Systems and Materials, MSM'2010, 5. bis 8. Juli 2009, Opole, Poland (pp. 184 – 185).
Seybold, L., Stania, M., Paczynski, A., Stetter R.: Intelligent Actuators for the Future of Individual Mobility. In: Proceedings of the 7th Workshop on Advanced Control and Diagnosis, ACD'2009, 19. and 20. November 2009, Zielona Góra, Poland.
Stania, M., Stetter, R.: "Mechatronics Engineering on the Example of a Multipurpose Mobil Robot". In: Solid State Phenomena Vols. 147-149 (2009) pp 61-66.
Stetter, R., Kleinmann, S.: Ganzheitliche Anwendung der Automatisierungstechnik für betriebswirtschaftlich nachvollziehbare Energieeffizienz. In: Economics Engineering, 11/2011.
Stetter, R., Paczynski, A.: Innovatives Lenksystem für Fahrzeuge für die flexible Produktion. In: Bericht über den Kongress Automation 2010 in Baden-Baden, Berlin: VDI, 2010.
Voos, H.; Stetter, R.: "Design and Control of a Mobile Exploration Robot". In: „Proceedings of Mechatronics 2006. 4th IFAC-Symposium on Mechatronic Systems". Heidelberg, 2006.
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.
Ward, S.: Electrical Engineering. Global Media: 2007.
Witczak, M.: Modelling and Estimation Strategies for Fault Diagnosis of Non–Linear Systems: From Analytical to Soft Computing Approaches. Lecture Notes in Control & Information Sciences. Berlin: Springer 2007.
Zając, M., Stetter, R., Uciński, D.: Particle Filter in Fault Detection for a Mobile Robot with an Innovative Drive System. In: Proceedings of CMM-2009 – Computer Methods in Mechanics, 18–21 May 2009, Zielona Góra, Poland.
Zaremba M. B.; Obuchowicz, A.; Banaszak, Z. A.; Rzejek, K. J. J.: A max-algebra approach to the robust distributed control of repetitive AGV systems. In: International Journal of Production Research, Volume 35, Number 10, 1 October 1997 , pp. 2667-2688 (22).
Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault–tolerant control systems. Annual Reviews in Control, 32, 229–252, 2008.
Ziemniak P.; Ucinski D.; Paczynski A.: "Robust Control of an All-Terrain Mobile Robot". In "Solid State Phenomena" Vols. 147-149 pp. 43-48, Switzerland, 2009.