Holistic Optimization of Tractor Management

Organic Computing in Off-highway Machines

Dipl.-Ing. **Timo Kautzmann**[1]; Timo.Kautzmann@kit.edu
Dipl.-Inform. **Micaela Wünsche**[2]; Micaela.Wuensche@kit.edu
Prof. Dr.-Ing. **Marcus Geimer**[1]; Marcus.Geimer@kit.edu
Dr.-Ing. **Sanaz Mostaghim**[2]; Sanaz.Mostaghim@kit.edu
Prof. Dr. **Hartmut Schmeck**[2]; Hartmut.Schmeck@kit.edu

Abstract

Recent developments in modern mobile machines like tractors introduce a large number of degrees of freedom for designers and engineers to optimize the system according to their individual goals. However, current machine management strategies generally use only a small part of the evolving potential, and the efficiency of machine operation is highly influenced by the individual skills of the operator. We present a novel approach to consider the mobile machinery as a whole and employ a holistic optimization to maximize efficiency. The optimized working point is the result of several on- and offline learning loops, which assign optimal actions to a set of possible situations. This is realized using an *Observer/Controller-architecture*.

Starting Point

Over the past decades, the development of mobile machines has been influenced by a rising amount of internal degrees of freedom [1]. As a consequence, mobile machines experience a rising complexity due to the higher amount of individual, cross-linked and communicating entities. Furthermore mobile machines and in particularly tractors execute a vast amount of different working cycles in varying environmental conditions.

This background poses a tremendous challenge for machine control by an adequate management strategy. In a conventional strategy, an operator sets basic defaults which are given as inputs into static characteristic curves or arrays in order to find optimized command variables for single control loops for some predefined cases. Although there has been much improvement towards automation and optimization especially due to the adoption of extensive electronics, working results are determined particularly by the experience and the skills of the operator [2].

[1]: Chair of Mobile Machines, Karlsruhe Institute of Technology (KIT), Gotthard-Franz-Str. 8, 76131 Karlsruhe, Germany
[2]: Institute of Applied Informatics and Formal Description Methods, KIT, Kaiserstr. 89, 76133 Karlsruhe, Germany

Goals

Without loss of generality, we select a tractor as a representative for mobile machines to maximize efficiency by holistic optimization. Efficiency is defined as ratio of power outputs at pulling devices, PTO and working hydraulics to heating value of fuel consumption. A holistic optimization process must consider interacting machine devices, environmental influences as well as different working conditions and drivers. Therefore, in a holistic management strategy, the machine must be to regard as depicted in Figure 1.

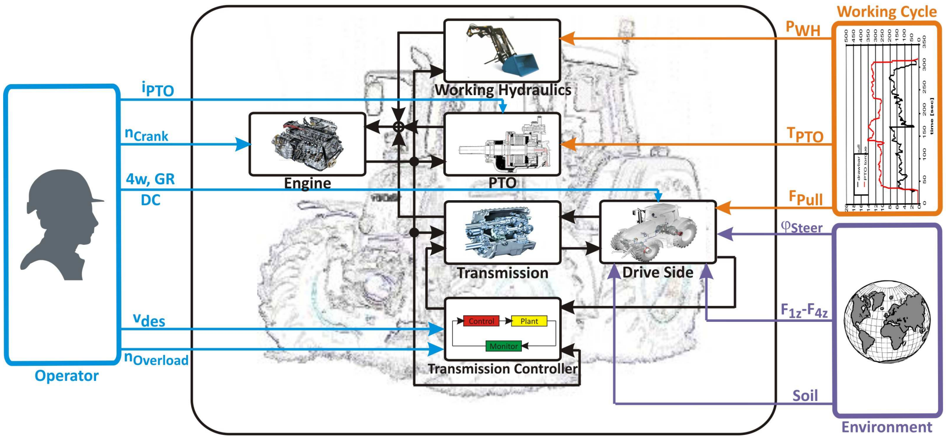

Figure 1: Tractor with external influences

According to that, a tractor consists of an engine, working hydraulics, PTO and the drive chain with transmission, drive side and transmission controller. The operator sets basic adjustments like PTO gear (i_{PTO}), crankshaft speed (n_{Crank}), 4 wheel clutch (4w), gear (GR), differential clutch (DC), desired velocity (v_{des}) and overload speed ($n_{Overload}$). The environment exchanges information about soil, vertical forces of single wheels (F_{1z}-F_{4z}) and steering angle (φ_{Steer}). The working cycle characterizes power flows of working hydraulics (P_{WH}), pulling force of coupling devices (F_{Pull}) and PTO torque (T_{PTO}).

Novel Approach

The main question when thinking about holistic optimization of a tractor management is: “How can we deal with so many complex and interacting devices and dependencies?” Complex interacting systems are widely spread in everyday’s life, e.g. traffic control, computer networks, electric power supply and automotive systems. A discipline that deals with the controllability of such self-organized systems is the field of *Organic Computing* [3]. Goal is to get a deeper understanding of the behavior of complex systems and develop design concepts to support an “organic” behavior that is characterized by so called self-X properties like e.g. self-organisation, self-optimisation, and the ability to learn.

A concept designed in Organic Computing to provide life-like attributes is the generic Observer/Controller(O/C)- architecture according to Figure 2 [4].

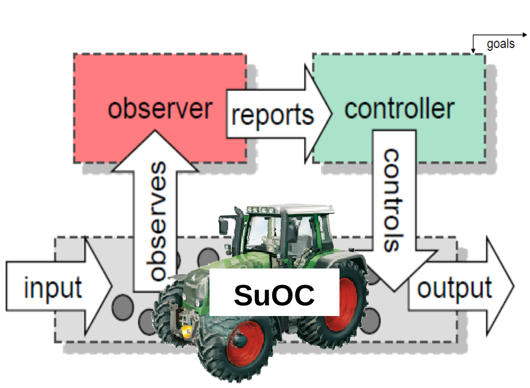

Figure 2: Generic Observer/ Controller architecture

This O/C-architecture is designed to holistically observe and optimize an underlying *System under Observation and Control* (SuOC) according to an externally given optimization goal. In this case, a Fendt Vario is to be optimized regarding its efficiency. The O/C-architecture observes the overall system and interacts only if a potential for improvement of the system is detected. In order to achieve this, the *Observer* characterizes the current system state, or situation, by evaluating sensor data from the SuOC. The system state is specified by the currently executed working cycle of the tractor and the environment according to Figure 1. The Observer aggregates all relevant data and reports it to the *Controller*. The Controller evaluates the system behaviour according to a given target function (system goal). Thereby the Controller learns adequate reactions for identified situation, so called action. The action is able to access all internal system parameters or degrees of freedom in the system. In conventional management strategies they are set by the operator according to Figure 1. Therefore, the current situation can be summarized to $\vec{v}_S$, the action to $\vec{v}_A$.

Current Work

Currently, the tractor as SuOC is simulated by a model in AMESim according to Figure 3. Input into the AMESim model is the *PowerMix* of the *German Agricultural Society (DLG)*, which describes the main working cycles a tractor performs. Output is efficiency.

The clustering module in the Observer measures the situation vector $\vec{v}_S$ and over time clusters the incoming vectors into groups. Each cluster is interpreted as a different system state, for which an adequate action should be taken. It is identified by a cluster ID (cl_ID) and represented by a cluster center (cl_center). The clustering is done by a density based clustering algorithm. In each time step, the current cluster ID is reported to the mapping module within the Controller, where an adequate action $\vec{v}_{A,opt}$ is assigned to each known cluster ID. This action will be sent as adjustment to the SuOC. If the cluster ID that has been reported by the Observer is yet unknown to the mapping, the offline learning cycle within the Controller is activated. Here, the adaptation module uses an evolutionary algorithm to generate candidate actions for the new situation, which are evaluated using an internal simulation model of the SuOC. It returnes for each candidate solution $\vec{v}_A'$ a simulated efficiency $\eta(\vec{v}_A')$ that is to be maximized. The best performing actions are sent back as $\vec{v}_{A,opt}(cl\_ID)$, which now enables the mapping to react to the new situation as soon as it arises again.

To evaluate the rules in the mapping, an online learning loop complements the model based offline learning. Therefore, the efficiency η_{real} of the SuOC is measured, and stored in a history. Every time the Controller executes an action $\vec{v}_{A,opt}$, average efficiency $\overline{\eta}_{real}$ before and after the execution are compared, in case $\vec{v}_S$ stays constant. If system efficiency is higher after the execution, an evaluation value (ev) of the corresponding rule will be increased, or decreased if it is lower. Rules that repeatedly show a bad performance will eventually be deleted from the mapping and replaced by new ones.

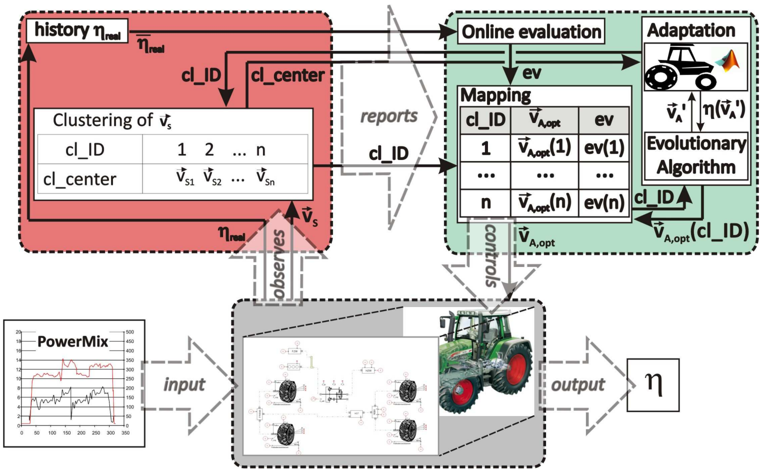

Figure 3: Specific Observer/Controller layout

Results

The architecture described above was tested with *PowerMix* cycle Z5K - rotary harrow. Characteristic working cycle graphs of PTO torque and pulling force as well as results of the clustering module are illustrated in Figure 4.

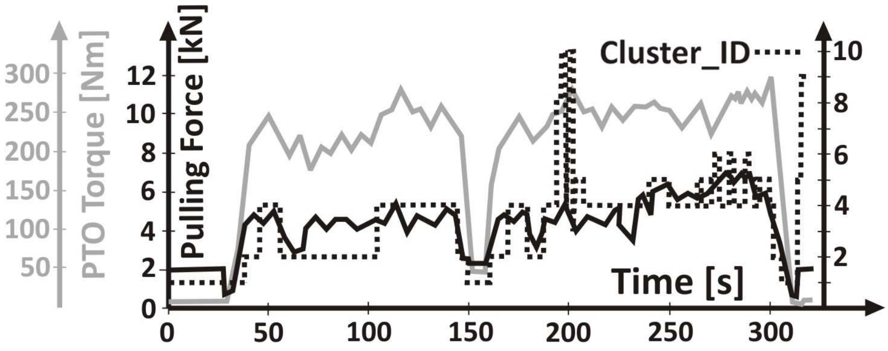

Figure 4: PowerMix cycle Z5K with indicated clusters

For a stable situation (steady cl_ID) between 60s and 100s the optimization process will be demonstrated. Since there is no $\vec{v}_{A,opt}(cl\_ID)$ in the mapping yet, the adaptation module generates new rules by means of an evolutionary algorithm over 100 generations and a population of 100 individuals. A comparison between efficiency of the original $\vec{v}_A$, and the newly learned $\vec{v}_{A,opt}(cl\_ID)$ that is returned by the Controller's internal model of the SuOC is shown in Table 1. It can be seen that within the Controller's internal simulation model, the suggested new action leads to a considerable increase in system efficiency.

Table 1: Comparison of original and suggested action

$\vec{v}_A$ (original)	$\vec{v}_{A,opt}$
$\begin{pmatrix} n_{Crank} \\ v_{des} \\ i_{PTO} \\ DC \\ GR \\ 4w \\ n_{Overload} \end{pmatrix} = \begin{pmatrix} 2100\ rpm \\ 1{,}4\ m/s \\ 0{,}2647 \\ open \\ 1 \\ closed \\ 1680\ rpm \end{pmatrix}$	$\begin{pmatrix} n_{Crank} \\ v_{des} \\ i_{PTO} \\ DC \\ GR \\ 4w \\ n_{Overload} \end{pmatrix} = \begin{pmatrix} 2100\ rpm \\ 8{,}2\ m/s \\ 0{,}4776 \\ closed \\ 1 \\ open \\ 1800\ rpm \end{pmatrix}$
$\eta(\vec{v}_{A,original}') = 9{,}75\ \%$	$\eta(\vec{v}_{A,opt}') = 17{,}9\ \%$

Figure 5 shows the results of applying the new rule to the real SuOC (in our case the AMESim model). In this particular situation, the O/C architecture is able to increase efficiency by about 50%.

However a $\vec{v}_{A,opt}$ like this one is not practicable e.g. due to the high v_{des}. In the current version of the implementation, the focus lies exclusively on the optimization of the target function (maximization of efficiency), in order to show the potential of the architecture as a machine management system for mobile working machines. The consideration of restrictions for entries in $\vec{v}_{A,opt}$, like, e.g. a minimum or maximum velocity, can be easily introduced by restricting the search space of the optimisation algorithm in the adaptation module accordingly.

Additionally noticeable is the difference between calculated efficiency of the internal model in the controller $\eta(\vec{v}_A')$ and average efficiency $\overline{\eta}_{real}$ of the actual SuOC. Reason for that is most likely the steady state approach of the controller model.

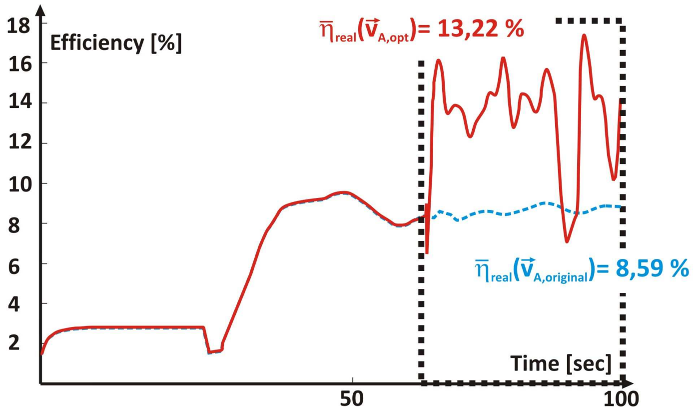

Figure 5: Real efficiencies of original and suggested action in Z5K

References

[1] Kautzmann, T. et. al.: A novel approach for holistic optimization of mobile machine managements. In proceedings of The Twelfth Scandinavian International Conference on Fluid Power SICFP'11, vol. 2, 2011, 207-221.

[2] Kunze, G. et. al.: Bedienereinfluss auf Leistungszyklen mobiler Arbeitsmaschinen. ATZ offhighway, Sonderausgabe April 2011, 70-79.

[3] H. Schmeck. Organic Computing - A New Vision for Distributed Embedded Systems. In Proceedings Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC 2005), pages 201–203. IEEE, IEEE Computer Society, 2005.

[4] J. Branke, M. Mnif, C. Mueller-Schloer, H. Prothmann, U. Richter, F. Rochner, and H. Schmeck. Organic computing - addressing complexity by controlled self-organization. In Post-Conference Proceedings of the 2nd International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2006), pages 185–191. IEEE, November 2006.