Heidelberg University
ZITI – Institute of Computer Engineering
Automation Laboratory

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

TITLE

Master Thesis

submitted by

Elena Elenkova

Matrikelnummer 328 00 12

conducted at

Automation Laboratory,
Heidelberg University

Date: 1st April 2016

Supervisor: Prof. Dr. sc. techn. Essameddin Badreddin
Second marker:

Declaration of Authorship

according to the § 16 (2) „Prüfungsordnung der Universität Heidelberg für den Master-Studiengang Angewandte Informatik " from 20 July 2010.

I have written the following thesis completely on my own without using any other sources and help materials other than the ones explicitly specified.

Mannheim, 1st April 2016

Elena Elenkova

Abstract

Contents

List of Figures

1 Introduction

1.1. Motivation

1.2. Goals

1.3. State of the art

The first hybrid vehicle was built in 1900 by Ferdinand Porsche. The car was called Lohner-Porsche Mixte Hybrid. It was a serial hybrid car with two wheel motors, had a 5.5 liter 18kW engine, a battery and an electric generator. The first mass produced hybrid car was the Toyota Prius in 1997 in Japan.

The first research papers about hybrid vehicles date back from the 1970s. LaFrance and Schult (1973) give an overview of a serial configuration and two parallel configurations - with a single and with a dual motor. They compare the suitability of different electric motor designs for various driving conditions.

Although there has been a lot of research in hybrid vehicles in the last two decades, there is a limited number of approaches which consider the power management problem from a game-theoretical point of view.

First of all, Gielniak and Shen (2004) describe a fuel cell hybrid electric vehicle and solve the power distribution problem as a two-player non-cooperative game. The vehicle has a fuel cell, battery, ultra capacitor and two 35 kW motors. The fuel cell tries to maintain a target 60% SOC of the Energy Storage Subsystem (ESS) or the battery. Power is firstly taken from the fuel cell and then the rest is taken from the ESS. The voltage of the ESS depends on the SOC and it is also being cooled. The

Ultra Capacitor (UC) is a lumped energy storage device, its SOC depends on the UC voltage and it is also air cooled. The motor has a maximum bus power limit. There is also a transmission model and accessory loads. The accessories such as air conditioner, power steering are powered from the electric motor.

The game-theoretical approach involves two players - all power supplying components as the first player and all power consuming components as the second player. The Master Power Management Controller (MPMC) governs all components in the power-train and calculates the solution. The objective of the game is to save fuel and to accelerate fast. The decisions of the players are how much power to supply at a given moment. The payoff of each player is represented by utility functions - efficiency, performance and composite utilities. The efficiency utility is a function of the strategy of the opponent - how much power they contribute. The performance utility is in this case acceleration. The composite utility is then computed for a time moment after all components have been mapped to an efficiency or performance function. The authors Gielniak and Shen (2004), however, do not provide any insights on how exactly the game-theoretical solution was computed.

Regarding simulation three drive-cycles were examined - Federal Test Procedure (FTP), US06 and Constant 60 Miles Per Hour. A simulation tool called ADVISOR (Burch, Cuddy and Markel, 1999) is used. A comparison is drawn between a basic control strategy without game theory and a game-theoretical control strategy. In the US06 cycle, the Game Theory control achieves more miles per gallon than the Basic control. For acceleration the Basic control is able to accelerate from 0 to 85 miles in 17.48s, whereas the Game Theory control needs only 11.77s.

A further approach towards power control is presented by Chin and Jafari (2010). The hybrid is based on the Toyota Prius and has a gasoline engine and one electric motor which are the two players in a bimatrix game. The vehicle configuration consists of a Gasoline Engine Controller, Electric Motor Controller and a Power Controller as the main unit for the computation of the solution. On the one hand, the Gasoline Engine Controller manages the fuel and air injection. A three-way catalyst system is utilized for the gas emissions of HC, CO_2, CO and NO_x. On the other hand, the Electric Motor Controller includes also battery and capacitor units. Batteries have high energy capacity but cannot provide much power, whereas capacitors have less energy capacity, but can produce a lot of power. Capacitors help improving battery lifetime and are also useful for providing short bursts of power during acceleration. The Power Controller is responsible for computing the game theory solution. It takes the requested torque from the driver and determines the optimal strategies for the engine and the electric motor. It outputs the torque for each of the two power sources. In addition, it charges the battery when the SOC is low. The goal of the Power Controller is to minimize fuel usage and at the same time maximize torque and reduce gas emissions. This hybrid configuration also has different modes of operation - Engine only, Motor only, Mixed and Battery Charge.

The game-theoretical solution consists of a non-cooperative bimatrix game. The payoff matrices for both players - the engine and the electric motor, are of size $M \times N$. These denote the strategies of player 1 and 2. The game is solved by a Nash Equilibrium (Nash, 1951). When player 1 chooses strategy $i \in M$ and player 2 chooses strategy $j \in N$ the payoff is (a_{ij}, b_{ij}) and it constitutes the Nash Equilibrium if this pair contains the optimal strategies for both players. Pure strategies are extended to mixed by specifying the strategy as a vector of probabilities over all pure strategies. The strategies are how much torque to contribute, measured from 0 to 6000rpm. Each payoff entry in the matrix is a function of fuel consumption, gas emissions, engine temperature, SOC deviation, extra weight and driver's demands. The Nash Equilibrium of the game is computed by the Lemke-Howson algorithm (Lemke and Howson, 1964) in the Power Controller. However, no simulation results are presented.

Dextreit and Kolmanovsky (2014) use the hybrid configuration of the Jaguar Land Rover Freelancer2. It has a diesel engine, two electric motors and a six-speed dual clutch transmission. The first electric motor is attached to the engine crankshaft integrated starter generator (CISG), while the second is attached to the rear wheels and is called the electric rear axle drive (ERAD). There are five driving modes. The EV mode is when only the ERAD provides the torque, the Engine-only mode is when only the engine supplies the torque. The parallel mode means that both the engine and the motors provide the torque. There is a charging mode in which the engine produces the driving torque and the CISC torque. The last mode is a boosting mode, where the engine supplies the additionally needed driving torque.

The game-theoretical approach Dextreit and Kolmanovsky (2014) adopt is a finite-horizon non-cooperative game, where the two players are the driver and the powertrain. The cost function penalizes fuel consumption, NO_x emissions and the battery SOC deviation. In a typical dynamic programming approach the cost is a function of a state vector, a control vector and a vector of operating conditions. The driver chooses the operating conditions - requested wheel speed and wheel torque. The powertrain chooses the control variables and the state vector is the SOC of the battery. The game is solved using a feedback Stackelberg equilibrium (Von Stackelberg, 1952). Firstly, the game is solved statically where the first player is the leader, who maximizes a function $J(w, t)$ by selecting the powertrain operating demands $w(t) \in W$. The second player (the follower) is able to observe the first player's decision and depending on that it selects its control vector $u(t) \in U$. It is assumed that both players make rational decisions. The pair (w^*, u^*) is the Stackelberg equilibrium. Next, the dynamic game is described which consists of $(T-1)$ stages, also called the horizon of the game. Given the initial state vector $x(0)$ the follower chooses their move $w(0)$. Then the follower selects their move $u(0)$. Thus, after the first stage the state is $x(1) = f(x(0), u(0), w(0))$. This continues until the last $(T-1)$ stage of the game. The GT controller Dextreit and Kolmanovsky (2014) implemented computes the state, control and operating values offline in three modules - GT Maps, Mode Arbitration and Mapping to torque demand. In the GT Maps module the wheel torque and speed, gear and battery SOC are discretized. The wheel torque is in the range of 0 to 1000 Nm. The wheel speed is between 0 and 100 rad/s. The battery SOC is between 40% and 70%. As output the module produces two modes to choose from. The Mode Arbitrator chooses one of these, for example the EV or the parallel mode. The aim of the Mapping to torque demand module is to distribute the torque between the engine and the motors.

Dextreit and Kolmanovsky (2014) present a baseline controller solved using Dynamic Programming and compare it with the game theory controller. Measurements are conducted over three drive-cycles for CO_2 emissions, equivalent to fuel consumption, NO_x emissions and deviation of SOC. These show that the GT controller outperforms the baseline controller.

There is another similar approach to the one of Dextreit and Kolmanovsky (2014). The idea is extended by Chen et al. (2014) who describe a single-leader multiple-follower game, where the follower is not only one (the battery) like in Dextreit and Kolmanovsky (2014), but also include other auxiliaries. The vehicle is a heavy-duty truck with an electric refrigerated semi-trailer and a battery. The aim is to manage the battery power and the refrigerated semi-trailer power. The vehicle model consists of the internal combustion engine. The total power which is requested from it is a sum from the driving power and the power for all auxiliaries. The semi-trailer model considers the air transfer from inside and outside and the cooling power. The problem is formulated as minimization of fuel consumption over a time interval. Energy balance is sought so that the battery energy and the air temperature in the semi-trailer at the end of the drive cycle are the same as in the beginning.

The game-theoretic approach assumes a single leader - the driver and multiple followers - all auxiliaries. The leader selects their action $w(t) \in W$ - the demanded wheel torque and speed. Then each follower f_i chooses their action $u_i(t)$. The game is split in two levels. The first level considers N two-player games between the leader and each of the N followers. It solves the games offline with a Stackelberg equilibrium where the driver maximizes and the leader minimizes the cost function. The strategies are stored in a lookup table for each follower. The second stage is computed online and combines all followers who play their Stakelberg strategies as stored in the lookup table. They all have to reach a mutual decision and this can be solved by a Nash Equilibrium. The central computation happens in the Energy Management System Operator (EMSO). It gathers the actions from all followers and sends them back the computed overall strategy for this stage. This happens until an equilibrium is reached. For simulation Chen et al. (2014) compare two control strategies - optimizing only the battery and optimizing both the battery and the semi-trailer power with the game theory approach. The optimized battery and trailer achieve better fuel consumption by 0.08%.

After exploring the literature, it can be concluded that all game-theoretical approaches assume a non-cooperative game and solve it either by a Nash Equilibrium or by a Stackelberg Equilibrium if the players are regarded as a leader and a follower. Mostly, two players are involved in the game methods proposed. In the leader-follower concept the players are the driver and the powertrain, as described in two of the approaches above. In another approach the two players are divided into the power-consuming and power-supplying devices in the powertrain. In another case Chin and Jafari (2010) described the two players as the electric motor and the engine. An important point is that no solution applies a cooperative game.

1.4. Content

2 Fundamentals

2.1. Game-theoretical solution approaches

This thesis examines a variety of solution approaches for cooperative games. They are Pareto Optimum, Nash Equilibrium, Nash bargaining solution, Kalai-Smorodinsky bargaining solution, the Core and the Shapley value.

2.1.1. Game definition

The generic game definition is described here. Additional definitions for each of the solution approaches are defined in the corresponding subsections of each solution.

A cooperative game is defined by the set of players N, where in this case there are two players - E and M for the engine and motor. It is also defined by the set of strategies of both players. Let us denote the number of pure strategies of each player with e and m. This makes it a bimatrix game, meaning that the payoffs of the game can be represented in two matrices of size $m \times e$. Let the payoff matrices A and B denote the payoffs for the first player, the engine, and the second player, the motor respectively, where $A = (a_{ij} : i \in \{1, ..., e\}, j \in \{1, ..., m\})$ and $B = (b_{ij} : i \in \{1, ..., e\}, j \in \{1, ..., m\})$. Sometimes player 1, E is called the row player and player 2, M is called the column player, since their strategies vary along the rows or along the columns of the matrices. The game is a non-zero sum game, since the sum of the two payoff matrices is not 0, $A + B \neq 0$. As opposed to a no-cooperative game, in cooperative games the players are allowed to make binding agreements among themselves in order to achieve a better payoff, that is they can form coalitions. We distinguish between the grand coalition of all players N or $\{1, 2\}$, where they all cooperate, from the individual coalitions $\{i\}, \forall i \in N$ which are $\{1\}$ and $\{2\}$. There is also the empty coalition, but it is not relevant. Each coalition has a value associated with it. The grand coalition forms its value as a sum of the payoffs of the engine and motor multiplied element-wise by a matrix with weights. The weights are distributed according to the torque deviation which the engine and motor achieve. The torque deviation is defined as the difference between the required and the actual torque. When the deviation is 0, the payoffs are weighted by 0.9, when it is between 0-10% of the required torque it is weighted by 0.91, when between 10-20% weight is 0.92 and so on up to 1.0 (the full sum of engine and motor torque).

The goal of the game is to save fuel and to maintain low gas emissions while achieving the required torque at any moment in time. Therefore, the payoffs have to be minimized. All of the solutions have been defined by taking this into account. Most of them work with maximizing payoffs, but for the purpose of this thesis their definitions and implementations have been modified to minimize the payoff function instead.

There exist two types of cooperative games - with transferable and with non-transferable utility. In transferable utillity (TU) games the payoff of one plyer can be transferred to another player without any loss. In contrast, in non-transferable utility (NTU) games the payoffs of each player cannot be redistributed among the other players. In this thesis the payoffs of the engine and the motor are not interchangeable, since decreasing the payoff of one player does not mean increasing the payoff of another at the same time.

The payoff functions are constructed in the following way. The engine payoff is:

$$\begin{aligned} a_{ij} = w_1 \times fuelConsumed + w_2 \times |requiredTorque - actualTorque| + \\ w_3 \times HC - emissions + w_4 \times CO - emissions + w_4 \times NOX - emissions + \\ w_6 \times SOC - deviation \end{aligned} \tag{2.1}$$

The motor payoff:

$$\begin{aligned} b_{ij} = 0.5 \times w_1 \times fuelConsumed + w_2 \times |requiredTorque - actualTorque| + \\ w_3 \times HC - emissions + w_4 \times CO - emissions + w_4 \times NOX - emissions + \\ w_6 \times SOC - deviation \end{aligned} \tag{2.2}$$

All in all, the game defined is a non-cooperative bimatrix non-zero-sum game with non-transferable utility.

2.1.2. Pareto Optimum

2.1.3. Nash Bargaining solution

Let us further define the game as a bargaining game, where each player

2.1.4. Lemke-Howson Algorithm

One of the most popular algorithms for finding a Nash Equilibrium for bimatrix non-zero-sum games is the Lemke-Howson algorithm (Lemke and Howson, 1964). The Matlab implementation developed by (Katzwer, 2014) was utilized to find one Nash Equilibrium in mixed strategies. This function contains a parameter, which affects the final result of the algorithm. Changing the parameter yields different Nash equilibria as output. This parameter k is the initial pivot, a number between 1 and $e + m$, where e and m are the number of strategies of player 1 and player 2 respectively. The general idea of the the algorithm is that it works on two graphs containing nodes and edges, one graph per player. It starts at the (0,0) point. It selects a k - the pivot, or the label of the graph, containing the strategy to be dropped first when traversing the graph. From there a path until the end is followed in order to find a Nash Equilibrium.

There a number of reasons for the different Nash Equilibrium solutions that the algorithm produces. According to Lemke and Howson, 1964 there exist an odd number of Nash Equilibria in a nondegenerate game. A nondegenrate game is a game where no mixed strategy with a support of size k has more than k pure strategies Nisan et al. (2007). Moreover, a support of a mixed strategy is defined as the set of pure strategies which have positive probabilities. Since there is an odd number of Nash Equilibria, there must be at least one Equilibrium, which proves that the algorithm will always find one solution in mixed strategies. However, which of all Nash Equilibria is found depends on which strategy label is dropped first. The initial pivot label which the algorithm drops can belong to either of the two players and be any of their e or m strategies.

As discussed in (Shapley, 1974) the Lemke-Howson algorithm possesses a significant weakness, namely that it is neither guaranteed that the algorithm will find all possible solutions, nor that it will tell if there are any unfound solutions.

The fundamentals of the Lemke-Howson algorithm are described next. Let us assume a scenario of a 2-player bimatrix game as the one used throughout this thesis, where the players E and M each have e and m number of pure strategies and their payoffs are in the matrices $A = (a_{ij} : i \in \{1, ..., e\}, j \in \{e+1, ..., e+m\})$ and $B = (b_{ij} : i \in \{1, ..., e\}, j \in \{e+1, ..., e+m\})$ respectively. The mixed strategies are the vectors $s = (s_1, s_2 ..., s_e)$ and $t = (t_{e+1}, t_{e+2}, ..., t_{e+m})$ where $S = s \geq 0; \sum_{i=1}^{e} s_i = 1$ and $T = t \geq 0; \sum_{j=e+1}^{e+m} t_i = 1$ are the sets for the mixed strategies spaces. Then, the payoff for player 1 is $\sum_{i=1}^{e} \sum_{j=e+1}^{e+m} a_{ij} s_i t_j$ and the payoff for player 2 is $\sum_{i=1}^{e} \sum_{j=e+1}^{e+m} b_{ij} s_i t_j$. An equilibrium is a pair of strategies (s^*, t^*) satisfying

$$\sum_{i=1}^{e} \sum_{j=e+1}^{e+m} a_{ij} s_i^* t_j^* = \max_{s \in S} \sum_{i=1}^{e} \sum_{j=e+1}^{e+m} a_{ij} s_i^* t_j^*$$

$$\sum_{i=1}^{e} \sum_{j=e+1}^{e+m} b_{ij} s_i^* t_j^* = \max_{t \in T} \sum_{i=1}^{e} \sum_{j=e+1}^{e+m} b_{ij} s_i^* t_j^*$$

If we also define the sets:

$$\tilde{S} = S \cup \{s \geq 0 : \sum_{i=1}^{e} s_i \leq 1 \text{ and } \prod_{i=1}^{e} s_i = 0\}$$

$$\tilde{T} = T \cup \{t \geq 0 : \sum_{j=e}^{e+m} t_i \leq 1 \text{ and } \prod_{j=e}^{e+m} t_i = 0\}$$

this allows us to define closed convex polyhedral regions S^i and S^j which together form S^k in $\tilde{S}$:

$$S^i = \{s \in \tilde{S} : s_i = 0\} \text{ for } i \in \{1, ..., e\}$$

$$S^j = \{s \in S : \sum_{i=1}^{e} b_{ij} s_i = \max_{l \in \{e+1, ..., e+m\}} \sum_{i=1}^{e} b_{il} s_i\} \text{ for } j \in \{e+1, e+2, ..., m\}$$

S^i contains all $\tilde{S} - S$ and S^j contains the mixed strategies for player 1 and the pure strategy j of player 2 which gives him the best response. S^i and S^j both make S^k and cover all of $\tilde{S}$. The same can be applied to define the regions T^k in $\tilde{T}$:

$$T^i = \{t \in T : \sum_{j=e+1}^{e+m} a_{ij} t_j = \max_{l \in \{1,...,e\}} \sum_{j=e+1}^{e+m} a_{lj} t_j\} \text{ for } i \in \{1, ..., e\}$$

$$T^j = \{t \in \tilde{T} : t_j = 0\} \text{ for } j \in \{e+1, ..., e+m\}$$

The Lemke-Howson algorithm represents the strategies of both players in two graphs with nodes and edges. The already mentioned definitions are required in order to define a labelling for the graphs. A labelling of a node consists of all of the labels of all surrounding regions of this node. Let the nonempty label of $s \in \tilde{S}$ be $L'(s) = \{k : s \in S^k\}$ and similarly the nonempty label of $t \in \tilde{T}$ be $L'(t) = \{k : t \in T^k\}$ and the label of the node pair with pure strategies $(s,t) \in \tilde{S} \times \tilde{T}$ be $L(s,t) = L'(s) \cup L''(t)$. A node pair (s,t) is completely labelled whenever $L(s,t) = K$, meaning that it contains the labels for all regions $k \in K$. A node pair is almost completely labelled if $L(s,t) = K - \{k\}$ for some $k \in K$.

Then, as in (Shapley, 1974) a node pair $(s,t) \in S \times T$ is an equilibrium point of (A,B) if and only if (s,t) is completely labelled. Let the two graphs be G' in $\tilde{S}$ and G'' in $\tilde{T}$. Two nodes are adjacent if they are on the two ends of the same edge which means that their labels differ in exactly one element. The set of k almost completely labelled nodes in the graphs and their edges are the disjoint paths of the graph and cycles. The equilibria of the game are always located at the end of these paths. Also, the starting node, by default (0,0) is called an artificial equilibria and it is also located at the end of a path.

The algorithm works by starting at $(s,t) = (0,0) \in G' \times G''$. Then a label k, which is to be dropped from (s,t), is chosen (initial pivot). This k can belong to either m or e. Let this node be (s,t) and its new label (after dropping k) be l. Then, if $l = k$ a Nash Equilibrium is reached. If $l \neq k$, then the algorithm continues by dropping another label and continuing until a completely labelled node has been reached, which is an equilibrium point. The starting label to drop is the pivot parameter as in the Matlab implementation. Since there are $m + e$ labels (strategies) which can be dropped first and they can end up in a different Nash Equilibrium node in the graph, it is inefficient to experiment with all different starting strategies. It will be assumed, that the default value 1 of the pivot as in the Matlab implementation of (Katzwer, 2014) is kept.

3 Application

4 Simulation

5 Implementation

6 Experiments

7 Discussion

Bibliography

Burch, Steve, Matt Cuddy and T Markel (1999). 'ADVISOR 2.1 Documentation'. In: *National Renewable Energy Laboratory.*

Chen, H et al. (2014). 'Game-theoretic approach for complete vehicle energy management'. In: *Vehicle Power and Propulsion Conference (VPPC), 2014 IEEE.* IEEE, pp. 1–6.

Chin, Hubert H and Ayat A Jafari (2010). 'Design of power controller for hybrid vehicle'. In: *System Theory (SSST), 2010 42nd Southeastern Symposium on.* IEEE, pp. 165–170.

Dextreit, Clement and Ilya V Kolmanovsky (2014). 'Game theory controller for hybrid electric vehicles'. In: *Control Systems Technology, IEEE Transactions on* 22.2, pp. 652–663.

Gielniak, Michael J and Z John Shen (2004). 'Power management strategy based on game theory for fuel cell hybrid electric vehicles'. In: *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th.* Vol. 6. IEEE, pp. 4422–4426.

Katzwer, Richard (2014). *Lemke-Howson Algorithm for 2-Player Games.* URL: http://www.mathworks.com/matlabcentral/fileexchange/44279-lemke-howson-algorithm-for-2-player-games (visited on 08/03/2016).

LaFrance, RC and RW Schult (1973). 'Electrical systems for hybrid vehicles'. In: *Vehicular Technology, IEEE Transactions on* 22.1, pp. 13–19.

Lemke, Carlton E and Joseph T Howson Jr (1964). 'Equilibrium points of bimatrix games'. In: *Journal of the Society for Industrial and Applied Mathematics* 12.2, pp. 413–423.

Nash, John (1951). 'Non-cooperative games'. In: *Annals of mathematics*, pp. 286–295.

Nisan, Noam et al. (2007). *Algorithmic game theory.* Vol. 1. Cambridge University Press Cambridge.

Shapley, Lloyd S (1974). *A note on the Lemke-Howson algorithm.* Springer.

Von Stackelberg, Heinrich (1952). *The theory of the market economy.* Oxford University Press.

A | Appendix A