Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Multi-axis Time Synchronization for Uncoordinated Motion Planning with Hard Constraints

Lukas Blaha*

**Department of Cybernetics, University of West Bohemia, Pilsen, Czech Republic, (karabina@centrum.cz)*

Abstract: This paper deals with multi-axis motion planning, especially with time synchronization of mutually independent axes of motions (uncoordinated motion). A novel time-optimal generator called GBAVS is used as a basic tool, utilizing the chain of integrators. The structure of generator uses 3 integrators for generating motion variables. The task is to plan the single-axis motions while respecting hard motion constraints and to achieve the final states in the same time. In contrast to other approaches the initial/final state does not necessary mean the state with zero velocity and acceleration, so called rest-to-rest motion. To reach synchronization, time scaling is based on decreasing the individual values of Jerk for each axis except the slowest one. Jerk is a parameter to be reduced as first. No initial/final state limitation and reduction of the value of Jerk primal to other motion limitations (e.g. velocity limits) are the main advantages of this approach.

Keywords: Trajectory generation, time-optimal control, time synchronization, multi-axis motion planning, hard constraints, Gröbner basis approach.

1 INTRODUCTION

Solving the problem of uncoordinated multi-axis motion planning is of importance for numerous practical applications, mostly there where is no restriction on motion path or where the supervision of admissibility is available. Minimizing the time of transfer between initial and final states is mostly required for this kind of applications. Typical examples are machine tools, pick and place machines or robots, where motion planning for each axis is made separately and appropriately adapted to the slowest axis.

Motion planning in single-axis is often done by time-optimal generator using chain of integrators. The standard task is to generate the motion variables from given initial to final state. Work in this area includes that of Nguyen et al.(2007) for rest-to rest motion, Kröger et al.(2006), Haschke et al. (2008) for general motion or some special cases. Other related works deal with generating the motion with hard motion constraints, mostly formulated as constraints on maximal velocity, acceleration, Jerk or even higher derivatives. Work by Kröger and Wahl (2010) focuses on online planning algorithms respecting the constraints. The constraints in this form naturally occur in real application as hard motion constraints. To reach the given transfer time for rest-to-rest motion, a simple algorithm for scaling the single-axis motion can be found. It follows from the integrator sequence. All motion limitations are scaled by a factor, given from time ratio. We get modified motion constraint parameters for single-axis generator. Resulting motion variables will then respect new lower limitations. For general motion, no similar time scaling approach exists, because of nonzero higher order derivatives of motion which have to be integrated. Moreover the modified limitations might get below a given initial/final state, which makes the problem unsolvable. Therefore the scaling is mostly done by lowering the velocities of each axis with using an adapting method, see Ezair et al. This paper uses earlier work of Bláha et al. (2009) and gives a simple procedure to time scaling of general motion. Slowing down the motion is primarily done by decreasing the Jerk limit, which is the main cause of motion inaccuracy, see work of Kyriakopoulos and Saridis (1988). Moreover this method does not affect the solvability, see details further.

2 TIME-OPTIMAL TRAJECTORY GENERATION

At first, we make the outline of trajectory generation in single-axis, described in earlier work of Bláha et al. (2009). Consider the trajectory generator in form of third order chain of integrators

$$\begin{bmatrix} \dot{s} \\ \dot{v} \\ \dot{a} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} s \\ v \\ a \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{B} u \tag{1}$$

where the state components are naturally labeled as

$$\begin{aligned} &s(t) - position \\ &v(t) - velocity \\ &a(t) - acceleration \\ &u(t) - jerk \end{aligned} \tag{2}$$

Next we consider the motion constraints in form

$$\begin{aligned} |v(t)| &\le VM \\ |a(t)| &\le AM \\ |u(t)| &\le BM \end{aligned} \tag{3}$$

Initial and final states are given as $x_0=[s_0, v_0, a_0]$, $x_f=[s_f, v_f, a_f]$ and have to respect the constraints (3), more precisely the convex subset, given by parabolic segments

$$v=\mp\frac{a^2}{2BM}\pm VM, \qquad (4)$$

see (Fig.1), and work of Bláha et al. (2009) in detail. The states are related by the expression

$$x_f=e^{At_f}x_0+\int_0^{t_f} e^{A(t_f-\tau)}Bu(\tau)\,d\tau. \qquad (5)$$

The task is to find the time-optimal control *u(t)* which satisfies (3), drives the system from its x_0 to x_f and minimizes the transfer time t_f.

It is well known, that using the Pontryagin Maximum Principle one finds that for the system (1) with bounded input, but without other constraints, the time-optimal control leads to ***bang-bang*** control with at most three time intervals, see Athans et al. (1966). For system with constraints and rest-to-rest motion planning is easy to understand, that the time-optimal control sequence for that system is ***bang-zero-bang*** type, with at most 7 time intervals, see the work of Castain and Paul (1984). To the best knowledge of the author, there is no exact proof of that, even though the complex constraints can be added to maximum Principle, see Locatelli (2001). This is why a following hypothesis will be accepted in the following parts of this work. The hypothesis uses the ***bang-zero-bang*** control strategy for general constrained system of chain of integrators and provides the possibility to convert the problem of time-optimal control of (1), (3) to a system of polynomial equations. The author believes that this hypothesis can be proved by using Maximum Principle with global instantaneous inequality constraints.

Hypothesis. *The time-optimal control for system (1) with state constraints (3) leads to* ***bang-zero-bang*** *control with at most seven time intervals* $\tau_1-\tau_7$ *where the input is constant. There are two possible strategies where the input alternates between +BM, 0 and -BM. Furthermore, single time intervals can vanish if the system does not reach the corresponding constraint. The general strategy of the time-optimal control can be expressed in two different forms*

$$u_{+(t)}=\begin{cases} BM, t\in[0,\tau_1) \\ 0, t\in[\tau_1,\tau_2) \\ -BM, t\in[\tau_2,\tau_3) \\ 0, t\in[\tau_3,\tau_4) \\ -BM, t\in[\tau_4,\tau_5) \\ 0, t\in[\tau_5,\tau_6) \\ BM, t\in[\tau_6,\tau_7) \end{cases}, \quad u_{-(t)}=\begin{cases} -BM, t\in[0,\tau_1) \\ 0, t\in[\tau_1,\tau_2) \\ BM, t\in[\tau_2,\tau_3) \\ 0, t\in[\tau_3,\tau_4) \\ BM, t\in[\tau_4,\tau_5) \\ 0, t\in[\tau_5,\tau_6) \\ -BM, t\in[\tau_6,\tau_7) \end{cases} \qquad (6)$$

where

$$\tau_k=\sum_{i=1}^{k} t_i, \qquad i=1..7\ , \ min\,\tau_7=min\sum_{i=1}^{7} t_i \qquad (7)$$

and t_i, *i=1,..,7 are the lengths of the intervals with constant input. From a given* x_0, x_f *and by searching all solutions in both sequences (6) is possible to clearly determine which solution transfers the system between the given states with minimal time, or whether the admissible control with respect the constraints does not exist.*

For detailed understanding we refer reader to work of Bláha (2010).

Note that if the constraints are equal to infinity or the system states do not reach them during the transfer, then the time-optimal control takes the well-known ***bang-bang*** form for third order system without state constraints.

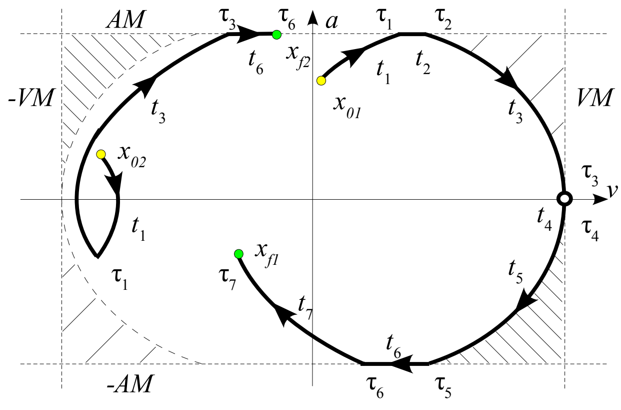

Fig. 1: Examples of state trajectories in *v-a* plane with depicted initial/final states, time intervals τ_i, and parabolic restrictions on state subspace.

2.1 Algebraic approach

The switching strategy uses algebraic approach, namely Gröbner basis approach. From Hypothesis there are only two strategies where input alternates as it is shown in (6), (7). Taking into account the maximal number of switching, we can find equations which define the trajectories from initial state to final state as some functions of time interval lengths t_i, *i = 1,..,7.*

We compose two sets of equations. One set for u_+, and one for u_-. For simplicity and because of symmetry of the task we will focus only on the case when u_+ is optimal. The strategy for u_- is analogous. From (5) and (6) we obtain the first three equations.

$$\begin{aligned} e_1:&\ a_f=a_0+BM(t_1-t_5+t_7-t_3) \\ e_2:&\ v_f=v_0+a_0(t_1+t_2+t_3+t_4+t_5+t_6+t_7) \\ &\quad +BM\Big[t_1(t_2+t_3+t_4+t_5+t_6+t_7) \\ &\quad -t_3(t_4+t_5+t_6+t_7)-t_5(t_6+t_7)-\frac{t_3^2}{2}-\frac{t_5^2}{2}+\frac{t_7^2}{2}+\frac{t_1^2}{2}\Big] \\ e_3:&\ s_f=f(s_0,v_0,a_0,t_1,t_2,t_3,t_4,t_5,t_6,t_7)_{\text{for lack of space}} \end{aligned} \qquad (8)$$

The possible active constraints define another four equations. If the acceleration constraint AM is active then

$$e_{4a}:\ t_1=\frac{AM-a_0}{BM} \tag{9}$$

otherwise

$$e_{4b}:\ t_2=0. \tag{10}$$

If the deceleration constraint -AM is active then

$$e_{5a}:\ t_7=\frac{-AM+a_f}{BM} \tag{11}$$

otherwise

$$e_{5b}:\ t_6=0. \tag{12}$$

Finally, if the velocity constraint VM is active then $v(\tau_3)=v(\tau_4)=VM$ and

$$\begin{aligned} e_{6a}:&\ VM=v_0+a_0\tau_3+BM\left(t_1t_2+t_1t_3-t_3^2/2+t_1^2/2\right)\\ e_{7a}:&\ VM=v_0+a_0(t_1+t_2+t_3+t_4)\\ &\quad +BM\left(t_1(t_2+t_3)+(t_1-t_3)t_4-t_3^2/2+t_1^2/2\right) \end{aligned} \tag{13}$$

else

$$\begin{aligned} e_{6b}:&\ t_4=0\\ e_{7b}:&\ t_{35}=t_3+t_5. \end{aligned} \tag{14}$$

As mentioned above, three are three types of constraint and each constraint can be active or inactive. Therefore we can compose 2^3 systems of seven polynomial equations $e_1,e_2,...,e_7$ for control sequence u_+. Finally we will get eight sets of seven equations for u_+ and eight sets for u_-. From hypothesis and restrictions (4) on x_0, x_f we know that one of sets of polynomial equations contains real positive solution of time-optimal control problem.

2.2 Algorithm GBAVS

Basic steps of computational algorithm are presented here. The main step uses the Gröbner basis theory to find all solutions of given polynomial set, first introduced by Buchberger (1986). Finding the Gröbner basis of given set can dramatically reduce the order of given basis polynomial equations and thus helps finding the solution easier. The key part of the algorithm is obtained using Gröbner basis approach. The step 3 of algorithm obtains explicit analytical equations for $t_1,...,t_7$. Therefore it is easy to directly evaluate the solution, except some polynomial root finding up to maximal 4^{th} order. This is done numerically, even that analytical formulas exist as well. Therefore the following algorithm provides analytic solution of given time-optimal problem.

input: x_0, x_f, BM, AM, VM

output: $t_1,...,t_7$, *control strategy* $u_\pm$

step 1. Test admissibility of $x_0=[a_0,v_0,s_0]$ and $x_f=[a_f,v_f,s_f]$. If x_0, x_f are in admissible domain then the solution must exist. Testing the vertices of transfer curve

$$v_{0\,\mathrm{lim}}=\pm v_0\mp\frac{a_0^2}{2\,BM},\quad v_{f\,\mathrm{lim}}=\pm v_f\mp\frac{a_f^2}{2\,BM}$$

to satisfy $VM\ge\{|v_{0\,\mathrm{lim}}|,|v_{f\,\mathrm{lim}}|\}$ and acceleration to satisfy $AM\ge\{|a_0|,|v_f|\}$.

step 2. Compose the eight sets S_i, $i = 1,2,..,8$ of equations for different possibilities of active constraints. Once for u_+ once again for u_- and insert the input data.

step 3. Evaluate all solutions from Gröbner basis for each set of equations S_i and take into account only real positive solutions.

step 4. Because all sets of equations do not intrinsically accept the constraints, check the limits for each time sequence

$$|a(\tau_1)|\le AM\,,|a(\tau_5)|\le AM\,,|v(\tau_3)|\le VM$$

step 5. If still exist more than one real positive solution, use that one which minimizes (7).

2.3 Example of trajectory generation

Assume that the system is already in motion with nonzero velocity and acceleration. We want to change these motion variables without changing the actual position for some reason. The input data could be following

$$\begin{aligned} &BM=0.2,\ AM=0.3,\ VM=0.7,\\ &x_0=[0,-0.5,0.2],\ x_f=[0,0.3,-0.1] \end{aligned}$$

The trajectory generator gives the lengths of time intervals corresponding to control sequence, specifically

$$\begin{aligned} &t_{1..7}=[0.2674,0,0.7674,0,0,1.2744,2.0000]\\ &u_-=[-BM,0,BM,0,BM,0,-BM] \end{aligned}$$

and output motion control sequences for motion states (2), see (Fig.2), and (Fig.3).

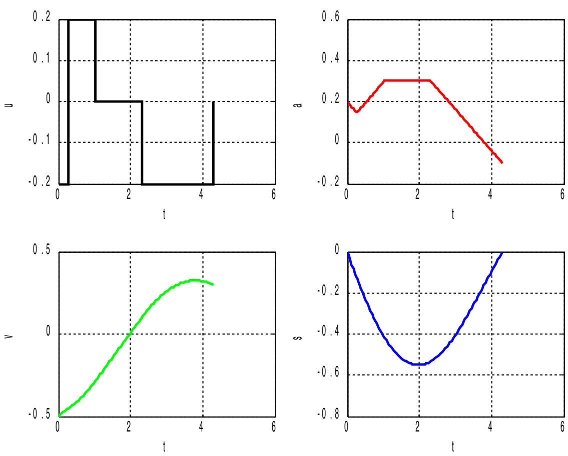

Fig. 2: Evolution of motion variables in time horizon to desired final state x_f.

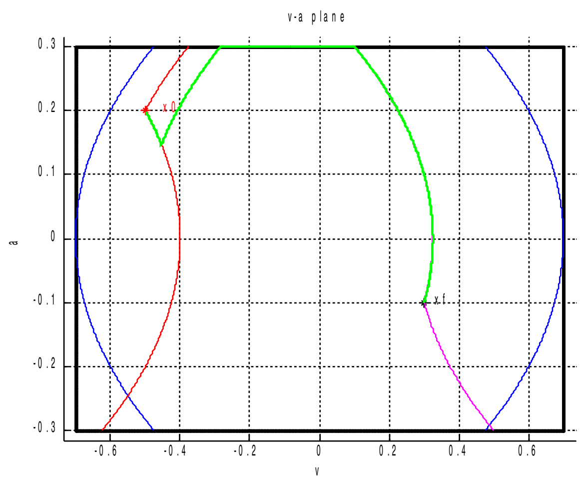

Fig. 3: Transfer trajectory of GBAVS generator in *v-a* plane, with depicted the initial/final states, switching curves, and admissible domain.

3 TIME SCALING FOR REST-TO-REST MOTION

Assume the trajectory generator in form of third order chain of integrators. Input parameters are, as in section 2 , the Jerk limit, acceleration limit, velocity limit and initial/final state BM, AM, VM, x_0, x_f. If the motion is considered as so called rest-to-rest motion, than we can easily scale the final transfer time using k-factor substitution in form

$$k=\frac{t_f}{\bar{t}_f},\quad \bar{BM}=BM\,k^3, \bar{AM}=AM\,k^2, \bar{VM}=VM\,k \tag{15}$$

where t_f is original transfer time, $\bar{t}_f$ is scaled transfer time given by the same control strategy with modified input parameters $\bar{BM}, \bar{AM}, \bar{VM}$. Unfortunately, this standard procedure obtains crucial restrictions on initial and final states. Let's give a simple deduction:

Assume the time scaling is in form (15). More precisely the actual state of generator has to follow the equation

$$x(t)=x(\bar{t}(t)) \tag{16}$$

where $\bar{t}$ is given by substitution $\bar{t}=t/k$.

Making the first derivation of (16)

$$\dot{x}(t)=\frac{\partial x(\bar{t})}{\partial \bar{t}}\frac{d\bar{t}}{dt} \rightarrow \dot{x}(t)=\dot{x}(\bar{t})/k \tag{17}$$

we get the k-factor scaling dependency for first motion derivative (velocity). This dependency implies that common initial/final conditions for $\dot{x}(t)$ and $\dot{x}(\bar{t})$ are due to k-factor multiplication only zero velocity conditions. Constraint on velocity *VM* is scaled by factor *k*.

Making the second derivation of (16)

$$\ddot{x}(t)=\frac{\partial \dot{x}(\bar{t})}{\partial \bar{t}}\frac{d\bar{t}}{dt} \rightarrow \ddot{x}(t)=\ddot{x}(\bar{t})/k^2 \tag{18}$$

we get the k-factor scaling dependency for second motion derivative. This again restricts the x_0, x_f to be zero in acceleration. The constraint *AM* is scaled by factor k^2.

Similarly for third derivation

$$\dddot{x}(t)=\dddot{x}(\bar{t})/k^3 \tag{19}$$

we get scaling by factor k^3 for Jerk limit.

K-factor scaling is therefore useful only for rest-to-rest motion. The shape of motion trajectories is not affected.

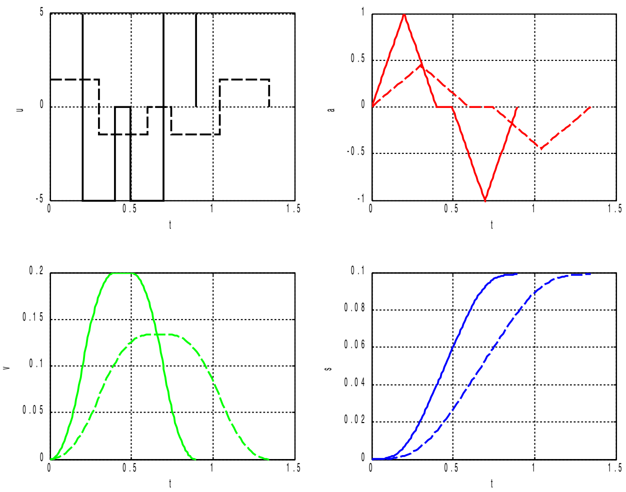

Fig. 4: Example of k-factor scaling, which changes all constraints, but not the shape of trajectories.

4 TIME SCALING FOR GENERAL MOTION

Assume the GBAVS generator, described in section 2 . This generator can find t-optimal motion between any admissible x_0, x_f. Then we can use the bisection method to find appropriate value of Jerk, which slow down the motion to desired transfer time $\bar{t}_f$. Scaling algorithm repetitively uses bisection and trajectory generator GBAVS to find the control sequence with given transfer time $\bar{t}_f$. If there is additional demand on lowering the other constraints of motion, it can be easily put to GBAVS inputs. There is no restriction on input data, except the solvability, see section 2 .

Scaling of general motion does not have to have a solution. For example, when actual velocity is too high and lowered Jerk too small for not over passing the velocity constraint during the stopping. Note, that presented scaling by Jerk modification does not conserve the trajectory type (shape), unlike k-factor scaling.

5 BASIC EXAMPLE

Consider a planar milling machine with three independent degrees of freedom, represented by two linear shifts and one rotational axis of milling cutter. The general task is to roll on the cutter to desired position with given final non-zero translational velocity in *x*-direction and with given milling cutter revolutions, see (Fig. 5). Input data could look as follows

axis x:
$BM=5, AM=1, VM=0.2, x_0=[0,0,0], x_f=[0.3, 0.1, 0]$

axis y:
$BM=5$, $AM=1$, $VM=0.2$, $x_0=[0,0,0]$, $x_f=[0.1,0,0]$

axis of revolution ϕ :
$BM=200$, $AM=1000$, $VM=1500$,
$x_0=[0,0,0]$, $x_f=[-,1256(12000\ rev/min),0]$

Technologically there is no need to control the cutter orientation, only speed of rotation. Therefore we can use second order trajectory generator (GAVS) for cutter axis.

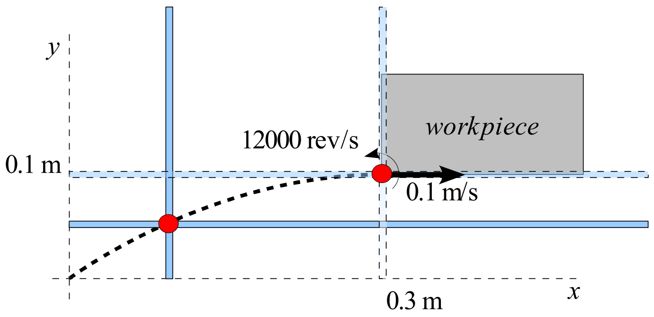

Fig. 5. The basic task structure with marked desired parameters of motion and presumed trajectory of motion.

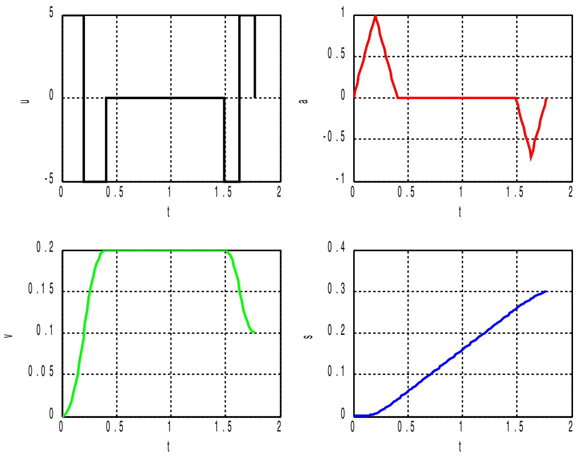

Fig. 6. Time-optimal motion trajectories of axis x before time-scaling.

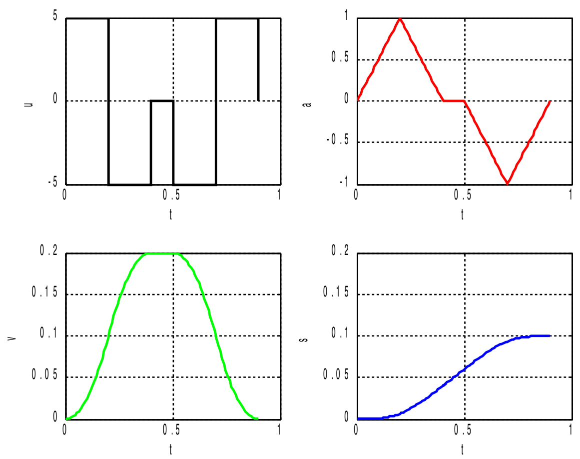

Fig. 9. Time-optimal motion trajectories of axis y before time-scaling.

GAVS generator is only simpler version of GBAVS, using only two integrators. We get motion trajectories, see (Fig. 8), and final transfer time $t_{f\phi}=5.01\text{s}$. For translational axes we use GBAVS and get transfer times $t_{fx}=1.77\text{s}$, $t_{fy}=0.9\text{s}$, see (Fig. 6) and (Fig. 9) for motion trajectories. Comparing

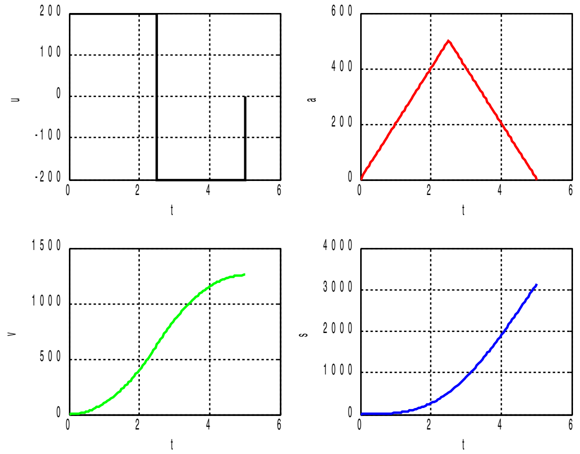

Fig. 8. Time-optimal motion trajectories of milling cutter

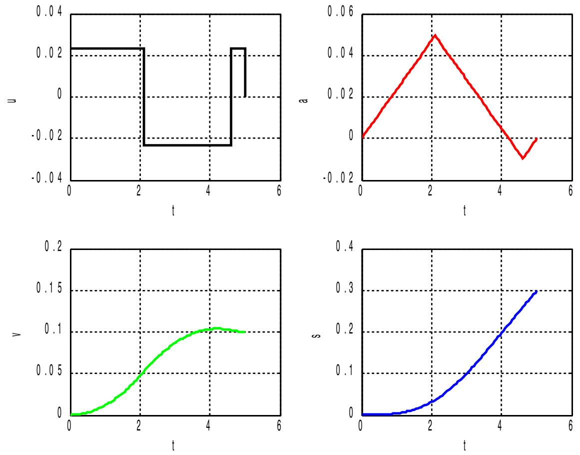

Fig. 7. Scaled motion trajectories of axis x for given final time.

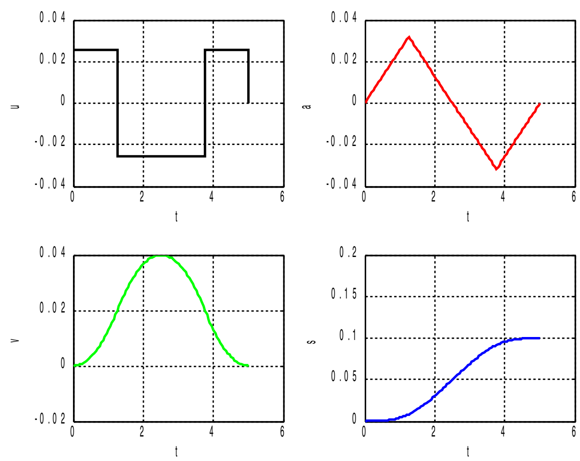

Fig. 10. Scaled motion trajectories of axis y for given final time.

the transfer times $\{t_{f\Phi}, t_{fx}, t_{fy}\}$, it is evident that we have to make a synchronization by extending the translational motions. Using procedure described in section 4 we get the modified input data for each axis, $\bar{BM}_x = 0.0235$, $\bar{BM}_y = 0.0253$ with preset accuracy. These data together with other given constraints and x_0, x_f leads to time-optimal trajectories with the same final time $t_{fx,y} = 5.01\text{s}$. (Fig. 7), and (Fig. 10) show the final motion trajectories.

CONCLUSION

In this paper, the time scaling problem for uncoordinated multi-axis motion planning has been solved. The hard motion constraints on each axis are fully respected. This solution uses third order trajectory generator called GBAVS in form of chain of integrators together with Gröbner Basis approach. This generator produces the time-optimal motion trajectories from arbitrary initial to arbitrary final state, fully respecting the constraints of motion. Time scaling is made according to slowest axis, by lowering only the Jerk limits. This ensures better shapes of motion trajectories, keeping the other limits unchanged, unless otherwise requested.

The presented approach has been validated through experiments. An example has been given in section 5 , where the main advantages of this approach have been shown.

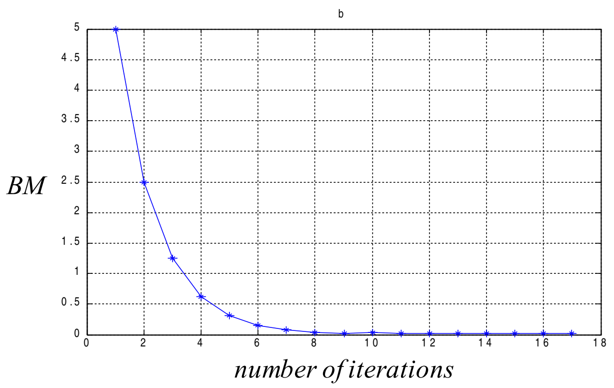

Fig. 11. Bisection Iterative convergence of Jerk parameter for x axis according to given final time. The time accuracy was preset to 10^{-2}.

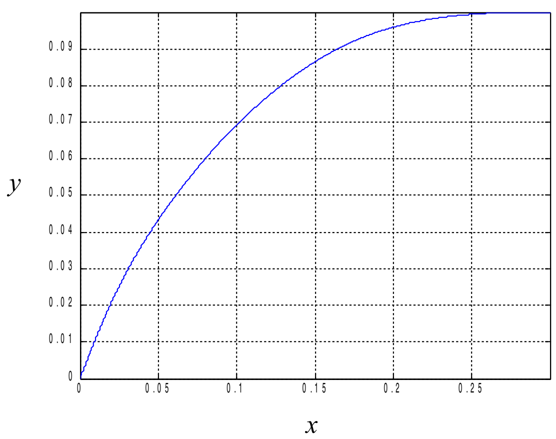

Fig. 12. Resulting (uncoordinated) motion of milling cutter in x-y plane.

ACKNOWLEDGEMENTS

The work was supported by the grant TA02020414 of Technological Agency of the Czech Republic.

REFERENCES

Athans, M. and Falb, P. (1966). *Optimal Control: An Introduction to the Theory and its applications*. New York: McGraw-Hill.

Bláha, L., Schlegel, M., and Mošna, J. (2009). Optimal control of chain of integrators with constraints. *17th International Conference on Process Control 2009*, Slovakia.

Bláha, L. (2010). Gröbner basis a control theory, *Thesis*, in czech language, University of West Bohemia, Pilsen.

Buchberger, B. (1986). Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory. *In Recent Trends in Multidimensional Systems Theory*, edited by N. K. Bose. chapter 6, 184-232. D. Reidel Publishing Company.

Castain, R.H. And Paul, R.P. (1984). An On-line dynamic trajectory generator, *Int. J. Robot. Res*., volume (22), 68-72.

Chang, S.S.L. (1963). Minimal Time Control with Multiple Saturation Limits. *IEEE Transaction on automatic control*.

Ezair, B., Tassa, T., and Shiller, Z. Multi-axis High-order Trajectory Planning.

Haschke R, Weitnauer E, Ritter H. (2008). On-Line Planning of Time-Optimal, Jerk-Limited Trajectories. *In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems.*

Kröger.T., Tomiczek. A., and Wahl,F.M. (2006). Towards On-Line Trajectory Computation. *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 736-741. Beijing, China.

Kröger.T. and Wahl,F.M. (2010). Online trajectory generation: basic concepts for instantaneous reaction to unforeseen events. *IEEE transaction on robotics,* volume (26(1)), 91-111.

Kyriakopoulos, K.J., Saridis G.N. (1988) Minimum jerk path generation. *Proc. {IEEE} International Conference Robotics and Automation,* 364-369.

Locatelli, A. (2001). *Optimal Control: An Introduction*. Basel, Boston, Berlin; Birkhäuser Verlag.

Nguyen, K.D., Chen, I.M., and Ng, T.C. (2007). Planning algorithms for s-curve trajectories. *IEEE/ASME international conference on Advanced intelligent mechatronics*.

Walther, U., Tryphon, T.G. and Tannenbaum, A. (2001). On the Computation of Switching Surfaces in Optimal Control: A Gröbner Basis Approach. *IEEE Transaction on automatic control*, volume (46), 534-540.