Procedia Manufacturing

Volume 1, 2015, Pages 216–225

43rd Proceedings of the North American Manufacturing Research
Institution of SME http://www.sme.org/namrc

# A Smooth Trajectory Generation Algorithm for Addressing Higher-Order Dynamic Constraints in Nanopositioning Systems

Akilan Bharathi, Jingyan Dong

*Department of Industrial and Systems Engineering, North Carolina State University, Raleigh, NC*
*jdong@ncsu.edu*

**Abstract**

The generation of a time-optimal feedrate trajectory has received significant attention in CNC machining and robotics applications. Most of the existing feedrate planning algorithms take velocity and acceleration into the consideration for capability constraints. The introduction of higher order dynamic states, such as jerk and/or jounce into the feedrate scheduling problem makes generating computationally efficient solutions while simultaneously guaranteeing optimality a challenging problem, as the dimension of the planning problem is increased accordingly. This paper proposes a heuristic trajectory planning algorithm that can provide a near optimal trajectory for problems with higher order dynamic states. The algorithm starts with a non-optimal but feasible velocity trajectory, which is interpolated from a number of knot points by piece-wise spline interpolation with high order continuity. Then the trajectory is improved by scanning the interpolating knot points and increasing the velocity at each knot points while maintaining the feasibility of the resulting trajectory. A near optimal trajectory is achieved when the improvement in travel time is neglectable from the last scan iteration. The algorithm supports the incorporation of high order dynamic states (up to fifth order derivative of position) in constraints for optimization without sacrificing the computational efficiency. Examples including linear and curved toolpaths are presented to illustrate the effectiveness of this algorithm for high-speed contouring.

*Keywords:* Feedrate Scheduling, Jerk Constraints, Trajectory Generation, Optimal Control

# 1 Introduction

Machining efficiency is critically important to the competitiveness of the modern manufacturing demands, which necessitates a successful approach to minimize the time required for a numerically controlled manufacturing system to perform a given task. The dynamic capabilities of the machine tools or robotics need to be fully utilized to achieve the optimal performance. The capabilities from the physical systems (*e.g.* actuators, servo systems) impose constraints on the feasible velocities,

Peer-review under responsibility of the NAMRI Scientific Committee
doi:10.1016/j.promfg.2015.09.006

accelerations, jerks of a multi-axis, numerically controlled (NC) machines. For a target trajectory on a specific multi-axis system, to identify an accompanying feedrate profile that can be executed in minimum-time without exceeding the capabilities of the machine systems, is a non-trivial optimization problem.

The off-line trajectory planning and feedrate optimization problem have been studied by many researchers. There are a number of trajectory planning algorithms which generate trajectory to drive a manipulator or motion system with certain joint torques or actuator forces capabilities along a given geometric path in minimum or near-minimum time (Shin 1986; Dong 2006; Bharathi 2014; Bobrow 1983; Dong 2007; Zlajpah 1996; Timar 2005). In most of these earlier works, either the system dynamics is simplified (for example, Coriolis and centrifugal terms are omitted in industrial robot dynamics) or hardware capability constraints are developed without higher-order states (*e.g.* jerk constraint (Dong 2007)). In these algorithms, the velocity limit curve (or boundary curve) is obtained from the system constraints, and a bang-bang type of trajectory is generated with alternating maximum acceleration and maximum deceleration segments. For this approach, higher-order terms and constraints cannot be readily incorporated into the existing algorithms as their control variables are acceleration. The other approach to address the minimum-time trajectory generation problem is to use the optimal control framework. The problem can be easily expressed as a traditional minimum-time problem with dynamic state and control dependent constraints and can be solved through the application of Pontryagin's minimum principle. A number of research works (Gourdeau 1989; Chen 1989; Shiller 1994; Tarkiainen 1993) have been expended to solve this minimum-time trajectory planning problem by studying the optimality conditions, avoiding singularities and generating numerical solutions. Although the optimal control framework does allow for a systemic formulation of the minimum-time problem incorporating higher-order constraints, the nature of this problem and the number of singular configurations one encounters in practical application make it very difficult to obtain accurate and efficient solutions. In manufacturing applications, the feedrate scheduling and optimization problem has been studied by several researchers in the past (Butler 1989; Imamura 1989; Farouki 2000; Imamura 1991). Researchers have developed algorithms with a two-pass structure (Renton 2000; Dong 2007; Bieterman 2002) to solve the minimum time feedrate optimization problem. These algorithms are computationally efficient compared to the phase plane approaches and optimal control methods. However, these prior approaches are still difficult to include torque rate or jerk constraints.

Incorporating higher-order constraints (*e.g.* jerk and/or jounce) is very important for many manufacturing machines. For example, without jerk constraints, the acceleration profile from the optimization has discontinuities, which correspond to step changes in the force output demanded of the drive, giving rise to large contouring errors, exciting vibrations in the transmission and bearing elements of the drives, producing noise during operation and, in general, accelerating wear in the system. These effects are even serious for flexure-based nanopositioning systems, due to their neglectable damping. Abrupt changes in acceleration can cause severe vibration and degrade the stability of the systems.

It is very difficult to introduce higher-order constraints into the existing algorithms, as the complexity and the dimension of the problem are greatly increased with the addition of higher-order constraints. In the phase plane approach, the analysis must now be performed in a three dimensional position-velocity-acceleration phase space instead of the position-velocity phase plane. The identification of appropriate switching points from the velocity-acceleration surface turn out to be extremely difficult. To overcome the above mentioned hindrance, researchers developed an acceleration continuation procedure that can be supplemented to the existing optimal algorithm framework to make the feedrate optimization algorithm address jerk constraints (Dong 2007). These algorithms add multiple passes to handle the arising infeasibilities in jerk profiles. Overall, these algorithms become very complex and computationally intensive making it impractical for large scale industrial applications. A few prior research tried to combine trajectory planning and spline interpolation approach to minimize cycle time while considering jerk limitations (Costantinescu 2000; Mattmüller 2009). In this approaches, a set of cubic splines were used to construct the trajectory in the phase plane. The knot-point locations were chosen to be the singularity point locations in the time-optimal motion without considering jerk. Such selection of the knot points again limits their application in large scale optimization problem.

This paper introduces a heuristic trajectory optimization algorithm that can time-efficiently provide a near optimal trajectory for problems with higher order dynamic states. The algorithm starts with a non-optimal but feasible velocity trajectory, which are interpolated from a number of knot points by piece-wise spline interpolation with high order continuity on position-velocity phase plane. Then the trajectory is improved by scanning the interpolating knot points and increasing the velocity at each knot points while maintaining the feasibility of the resulting trajectory. A near optimal trajectory is achieved when the improvement in travel time is neglectable between the current and previous scan iterations. The algorithm supports the incorporation of a variety of high order state-dependent constraints without sacrificing the computational efficiency. Several examples including linear and curved toolpaths are tested on a high-bandwidth multi-axis nanopositioning stage (Polit 2011), and presented to illustrate the effectiveness of this algorithm for high-speed contouring.

# 2 Feedrate optimization With High Order Constraints

The minimum time feedrate optimization problem can be applied to a number of applications from manufacturing to robotics with some uniform problem formulations. The geometric tool-path to be followed by each individual positioning axis is given as a parametric curve in the Cartesian space *i.e.* the axis positions $x^i$ are given by *eq.* (1),

$$x^i = P^i\left(u\right) : 0 \le u \le u_{\max} \ and \ 1 \le i \le n \qquad (1)$$

Where, *u* is a scalar parameter used to specify the geometric paths and *n* is the number of axis in the multi-axis positioning system. The points in the parametric space are usually interpolated using common interpolation techniques (*e.g.* splines or linear interpolation) to obtain a smooth path in the axis/joint space. The parameter *u* is sufficient to represent all the axis positions and hence referred to as parametric position variable. The parametric velocity $\dot{u} = du/dt$ and parametric acceleration $\ddot{u} = d^2u/dt^2$ and parametric jerk $\dddot{u}$ can be expressed from parametric position *u*.

In feedrate optimization, we wish to move the positioning platform or the machine tool as fast as possible along a certain path while obeying various system constraints. The capability constraints such as actuator torque limitations have been widely used in the robotics and manufacturing applications, and high order jerk constraints are very important to reduce mechanical vibration and improve the tracking accuracy. For the parametric toolpath given in *eq.* (1), the velocity, acceleration, and jerk of each axis can be expressed in the parametric terms as shown in *eq.* (2-4).

$$\mathbf{v}\left(\dot{u},t\right) = \frac{d}{dt}\mathbf{P}\left(u\left(t\right)\right) = \frac{d\mathbf{P}\left(u\right)}{du}\dot{u}(t) \qquad (2)$$

$$\mathbf{a}\left(\dot{u},\ddot{u},t\right) = \frac{d}{dt}\mathbf{v}\left(t\right) = \frac{d^2\mathbf{P}\left(u\right)}{du^2}\dot{u}(t)^2 + \frac{d\mathbf{P}\left(u\right)}{du}\ddot{u}(t) \qquad (3)$$

$$\mathbf{j}\left(\dot{u},\ddot{u},\dddot{u},t\right) = \frac{d}{dt}\mathbf{a}\left(t\right) = \frac{d\mathbf{P}\left(u\right)}{du}\dddot{u}(t) + 3\frac{d^2\mathbf{P}\left(u\right)}{du^2}\dot{u}(t)\ddot{u}(t) + \frac{d^3\mathbf{P}\left(u\right)}{du^2}\dot{u}(t)^3 \qquad (4)$$

The actuator torque limitations for DC motors are independent of displacement at all times, which results in parallelogram performance envelope in velocity-acceleration plane (Renton 2000). For nanopositioning systems, the special characteristic of the piezoelectric actuator is that the maximum actuator force is dependent on the instantaneous position *i.e.* its instantaneous state of compression/expansion (Bharathi 2014). From the trajectory planning point of view, we are interested in the relationship between the constraints of the physical system and the corresponding feasible dynamic states including displacement, velocity and acceleration.

All of above system dynamic state constraints can be expressed explicitly in terms of the parametric velocity, acceleration, jerk and even higher states as well as the path geometry. The constraints may be expressed in the general form,

$$C\left(u,\dot{u},\ddot{u},\dddot{u},\ddddot{u},\ldots\right)\leq 0 \qquad (5)$$

The problem can now be stated more formally as follows:

$$\min_{\dot{u}(t)} T=\int_{0}^{u\max}\frac{du}{\dot{u}(t)} \qquad (6)$$

Subject to

$$C\left(u,\dot{u},\ddot{u},\dddot{u},\ddddot{u},\ldots\right)\leq 0\,, \qquad \text{For all } u\in\left[0,u_{\max}\right].$$

In System (6), a parametric velocity function $\dot{u}(t)$ is to be determined such that the entire trajectory is executed in minimum time *T*. The constraints specify the system requirements that must be observed at all times. The minimum time trajectory planning problem is then to maximize $\dot{u}(t)$ so that the velocity trajectory should always be pushed upwards at the maximum value without violating the constraints specified by the system requirements.

# 3 Heuristic Smooth Feedrate Optimization Algorithm

Most of the previous feedrate optimization algorithms are very difficult to be extended for problems with higher order constraints, such as jerk and jounce. In this work, we developed a heuristic feedrate optimization algorithm that can incorporate any constraints without any limitation on higher order dynamic states. If we look at an optimal feedrate profile, it will be a continuous trajectory on the $u-\dot{u}$ phase plane. At each location of *u*, the parametric velocity $\dot{u}$ is maximum while all the constraints are satisfied. The high order dynamic states are constrained by the continuity and change-rate of the resulting trajectory. For example, a $C^0$ continuous trajectory on the $u-\dot{u}$ phase plane results in discontinuous acceleration profile and infinite jerk. A $C^4$ continuous trajectory on the $u-\dot{u}$ phase plane results in continuous acceleration, jerk, jounce, and derivative jounce profiles. The capabilities on these dynamic states (*i.e.* acceleration, jerk, jounce, and derivative jounce) of a physical system can be constrained by the rate change of the trajectory on the phase plane.

In our heuristic feedrate optimization algorithm, a set of knot points in the $u-\dot{u}$ phase plane are selected. The locations of the knot-points $\left(u_i,\dot{u}_i\right), i=0\ to\ n$ are fixed with equal increment of parametric position, but these knot points are also capable to be distributed differently with non-uniform distance from each other. The fifth order splines are interpolated from these knot-points. The optimization algorithm is to select the maximum $\dot{u}_i$ at each knot point position $u_i$ under the given constraints of the problem. Below a formal algorithm based on polynomial spline trajectory interpolation and point-by-point improvement is provided that can efficiently provide near optimal solution for the feedrate optimization problem.

**Algorithm: Heuristic Smooth-Trajectory Improvement Algorithm**

1. Initialize $u_0=0,\dot{u}_0=0,u_n=0,\dot{u}_n=0$ and set all parametric velocities to small values which are feasible *i.e.* set $\dot{u}_i=0$ for all the *n-1* knot-points *i = 1 to n-1*.

2. For the $i^{th}$ knot-point, find the maximum $\dot{u}_i$ by binary search with constraints satisfied by the neighboring interpolated trajectory segments.

   **Maximize** $\dot{u}_i$

   Subject to: $C\left(u_k, \dot{u}_k, \ddot{u}_k, \dddot{u}_k, \ddddot{u}_k, ...\right) \le 0$ for *k=i-p to i+p,* where *p* is the number of segments affected by the current improvement.
3. Set the parametric velocity for $i^{th}$ knot-point $\dot{u}_i = 0.63\left(\dot{u}_i\right)_{\max}$ according to the golden section segmentation technique, so as to give more room for the best improvement performance for the following knot-points.
4. $i = i+1, u_{i+1} = u_i + \Delta u$
5. If *i<n*, go step 2.
6. When *i=n*, calculate the total path traversal time from the current velocity trajectory. If the improvement in traversal time, in comparison to the previous scan, is less than the set *time-improvement* tolerance, the algorithm terminates. Otherwise, set *i=2* and repeat the steps 2 to 6 until the path traversal time cannot be further improved.

In this algorithm, for each knot-point, a binary search technique is used to find the maximum velocity for the current scan to improve the traversal time. When the improvement of the parametric velocity $\dot{u}_i$ for a known value satisfies all the constraints, the step-size of the next improvement trial will double the value of the previous step. Similarly, if the current improvement of parametric velocity causes the violation of any constraints, then the subsequent trial for improvement will be half of the value of the previous step. The search procedure will stop when the step of the velocity improvement is less than a given tolerance.

In the binary search for the optimal velocity at each knot-point, we do not need to validate constraints for the entire trajectory. When the parametric velocity of the current knot-point is under improvement, the change in the velocity only locally varies the interpolated trajectory on phase plane. As shown in Figure 1, as the parametric velocity at one knot point is changed significantly by 50%, only about 2 to 3 segments before and after the current knot-point of the interpolated trajectory have been significantly affected. The trajectory that is far from the current knot-point is not affected at all. Thus we only need to check the constraints for a few neighboring trajectory segments when optimizing the velocity for current knot-point, which improves the efficiency of the algorithm.

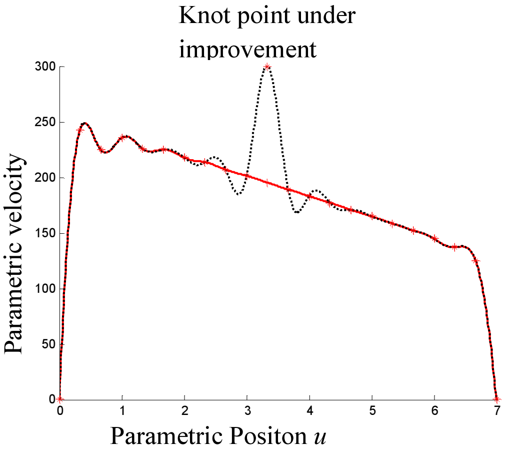

Figure 1: Change parametric velocity in one knot-point only affect the interpolated trajectory locally with a few segments around the knot-point.

When the optimal velocity $\left(\dot{u}_i\right)_{\max}$ is found for each knot-point, a smaller velocity $\dot{u}_i = 0.63\left(\dot{u}_i\right)_{\max}$ is set for trajectory interpolation, so that every knot-point has enough freedom to be uniformly pushed up towards its near-optimal value. Without such retraction, the "greedy" selection of the local velocity will limit the improvement of the next knot-point, as any small perturbation from the interpolation may make the trajectory around the knot-point infeasible.

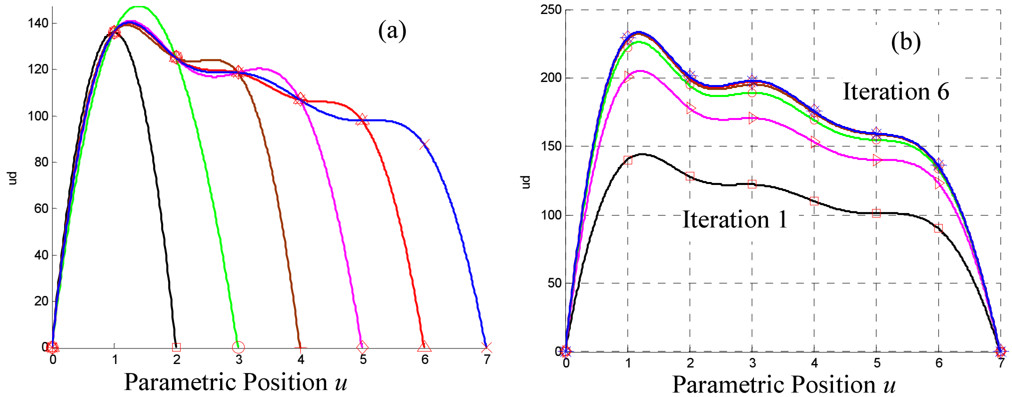

Figure 2: (a) Point to point improvement in the first scan iteration generates a feasible trajectory with reduced traverse time. (b) After six iterations, a near optimal trajectory is achieved.

Figure 2 demonstrates the typical optimization procedure for a linear toolpath. Six knot points are used to interpolate the trajectory on the $u-\dot{u}$ phase plane. In this algorithm, in one trajectory improvement iteration, all the knot-points are improved one by one in sequence. The improvement on each knot-point will push up the feedrate trajectory and reduce the traverse time of the toolpath, while keeping the feasibility of the resulting trajectory (Figure 2(a)). At the end of one improvement iteration/scan, we obtain a feasible trajectory whose traverse time is improved from the trajectory by the previous iteration. In this optimization framework, the maximum parametric velocity $\dot{u}_i$ for the $i^{th}$ knot-point is coupled with the parametric velocities of neighboring knot-points in trajectory interpolation. After all the knot-points are improved in one scan, the trajectory will have further space to be pushed-upward to reduce the travel time. Figure 2(b) shows the resulting trajectory from Iteration 1 to Iteration 6. A near optimal trajectory is achieved when the improvement in traverse time is neglectable from the last two iterations. The results from all six improvement iterations are listed in Table 1.

**Table 1: Parametric velocity at each Knot-Point from different Improvement iterations**

| Knot-Points | Scan #0 | Scan #1 | Scan# 2 | Scan #3 | Scan #4 | Scan #5 | Scan #6 |
|---|---|---|---|---|---|---|---|
| $u_d$ at P0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $u_d$ at P1 | 0 | 136.00 | 197.23 | 219.79 | 226.88 | 228.81 | 229.45 |
| $u_d$ at P2 | 0 | 125.04 | 175.31 | 194.00 | 199.16 | 200.45 | 200.45 |
| $u_d$ at P3 | 0 | 118.59 | 167.58 | 187.56 | 194.65 | 196.58 | 197.23 |
| $u_d$ at P4 | 0 | 106.99 | 150.82 | 168.22 | 174.67 | 176.60 | 176.60 |
| $u_d$ at P5 | 0 | 97.97 | 137.93 | 153.40 | 157.91 | 159.20 | 159.20 |
| $u_d$ at P6 | 0 | 87.67 | 121.82 | 132.77 | 135.00 | 136.64 | 136.64 |
| $u_d$ at P7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total traverse time (sec) | 0 | 0.1240 | 0.0879 | 0.0801 | 0.0782 | 0.0773 | 0.0772 |

In this paper, we have used a piecewise 5th order polynomial spline interpolation to arrive at a $C^4$ continuous trajectory. Hence constraints up to the order of derivative of jounce (fifth-derivative of displacement) can be readily incorporated into the algorithm without any change to the framework of algorithm or problem formulations. The heuristic smooth-trajectory generation algorithm in this paper, unlike other optimization approaches, requires little memory and as with any heuristics algorithm, the computational expense is much less compared with other optimization algorithms. For a set time-

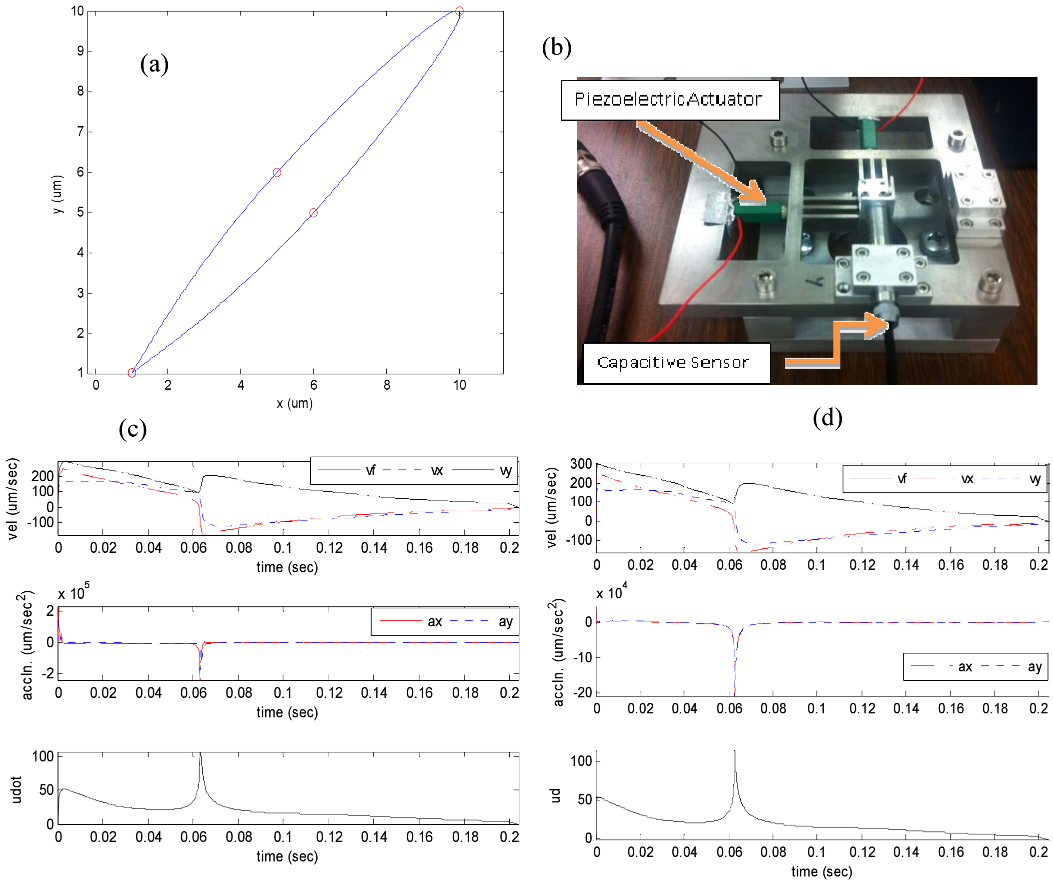

Figure 3: (a) An air-foil toolpath is used to compare the results from the heuristic trajectory improvement algorithm and bi-directional scan algorithm. (b) High-speed nanopositioning XY stage as the testbed. (c) Velocity, acceleration and parametric velocity profiles from bi-directional scan algorithm. (d) Velocity, acceleration and parametric velocity profiles from heuristic trajectory

improvement tolerance, the run-time for the algorithm has a linear relationship with respect to the length of the path and the number of trajectory optimization knot-points.

# 4 Validation and Case Studies

To validate the near-optimality of the heuristic trajectory improvement algorithm, we compare the result from the algorithm in this paper with an optimality-proven bi-directional scan optimization algorithm developed in (Dong 2006, 2007). In the bi-directional scan algorithm, a forward scan of the toolpath was used to identify the largest feasible value for parametric velocity at each trajectory point, and then a reversed scan is performed with the additional constraint that $\dot{u}$ should be lower than in the forward pass. By proving the local optimality of each trajectory segment and applying the principle of optimality, the global optimality of the resulting trajectory was demonstrated in (Dong 2006).

To make a fair comparison, the test path is taken to be the same cubic-spline air-foil shaped path shown in Figure 3a. Both the heuristic trajectory improvement algorithm and bi-directional scan

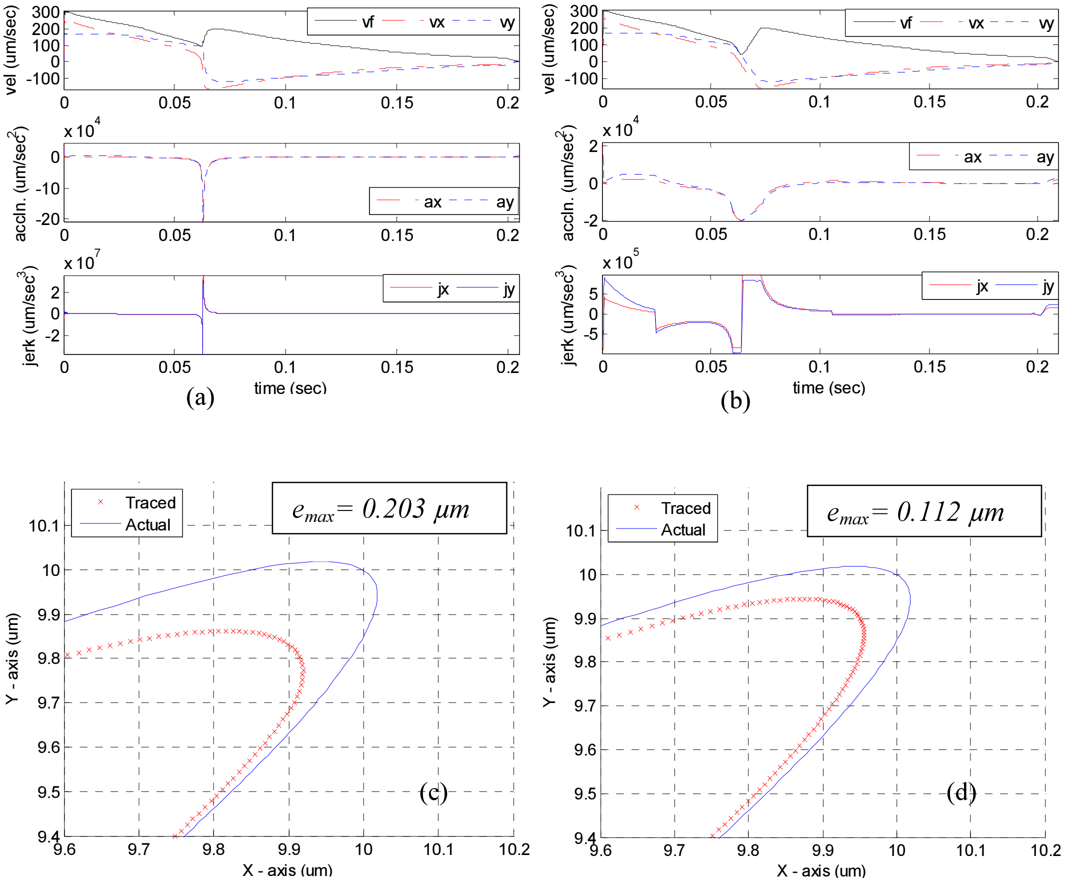

Figure 4: Comparison of tracking performance for air foil toolpath with and without jerk constraints. (a) Velocity, acceleration and jerk profiles without jerk constraints. (b) Velocity, acceleration and jerk profiles with jerk constraints. (c,d) Comparison of contouring accuracy between trajectories without (c) and with (d) jerk constraints.

algorithm were run subject to actuator performance constraints for a nanopositioning stage as derived in (Bharathi 2014, 2014). The two algorithms produce almost the same trajectory as shown in Figure 3c and 3d. Compared to the optimal results in traverse time from bidirectional scan algorithm (0.2039sec), the results for heuristic trajectory improvement algorithm (0.2048sec) is only less than 0.5% worse than the optimal results. When the time-improvement tolerance (stop criteria) was selected as 0.01% of total improvement, it took a total of 10 scans to arrive at this near-optimal result.

Besides the near optimal result provided from the heuristic trajectory improvement algorithm, it has much better expandability than bi-directional scan algorithm. In bi-directional scan algorithm, only velocity and acceleration related constraints can be added into the optimization problem. However, in the heuristic trajectory improvement algorithm, dynamic states up to the fifth-derivative of displacement can be easily applied into the optimization algorithm without changing the algorithm structure and increasing the complexity of the algorithm.

Figure 4 compares the optimization results without and with higher order jerk constraints for the same airfoil toolpath. The maximum jerk of the trajectory is limited to 1e+006 um/sec$^3$.Adding the jerk constraints only slightly increase the computation load of the algorithm. Only a few more iterations are added to generate the final trajectory. For the same toolpath, the execution time of the resulting trajectory with jerk constraints is increased slightly to 0.2093 sec from 0.2048sec without using jerk constraints. From Figure 4(a) and (b), the maximum jerk was successfully limited from very large value to the constrained value, especially at the sharp traverse corner of the toolpath. By limiting the maximum jerk experienced by the trajectory, significant improvement in the contouring accuracy can be achieved, and the nanopositioning system can be used with enhanced stability.

In this study, we used a high-bandwidth parallel kinematic stage (Polit 2011) as a test platform to carry out experimentation and obtain experimental data. For the airfoil tool path (Figure 4c and 4d), the maximum contour error at the sharp corner has been reduced by 45% from 0.203 μm to 0.112 μm, in comparison to the trajectory without limiting the jerk.

# 5 Conclusion

This paper presents a computationally efficient heuristic algorithm for feedrate optimization that can generate near optimal trajectory for a toolpath incorporating constraints with higher order dynamic states (*e.g.* jerk, jounce, and derivative of jounce). Instead of optimizing the trajectory along the toolpath, in our algorithm, the trajectory is constructed from a set of knot points on position-velocity phase plane by piece-wise spline interpolation with high order continuity. The algorithm starts with a non-optimal but feasible velocity trajectory, and the trajectory is improved in its traverse time by multiple scanning iterations to move the velocity profile upward. In each iteration, the velocity at the knot points is improved one by one while maintaining the feasibility of the resulting trajectory. A near optimal trajectory is achieved when the improvement in travel time is negligible from the last two scan iterations. The developed heuristic algorithm supports the incorporation of constraints with higher order states without sacrificing the computational efficiency. Examples including curved toolpaths were presented to validate the effectiveness of this algorithm in nanopositioning applications.

# Acknowledgement

This work was supported in part by the National Science Foundation under Grant Award NSF CMMI-1233176.

# References

Bharathi A and Dong J. Feedrate optimization and trajectory control for micro/nanopositioning systems with confined contouring accuracy. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 2014: doi: 10.1177/0954405414548467.

Bharathi A and Dong J. Time-Optimal Feed-Rate Scheduling for Nanopositioning Systems with Confined Contouring Error. In *Proceedings of NAMRI/SME North American Manufacturing Research Conference*, Vol. 42, June 2014.

Bieterman MB and Sandstrom DR. A curvilinear tool-path method for pocket machining. In *ASME 2002 International Mechanical Engineering Congress and Exposition*, pp. 149-158. American Society of Mechanical Engineers, 2002.

Bobrow JE, Dubowsky S, and Gibson JS. On the optimal control of robotic manipulators with actuator constraints. In *American Control Conference, 1983*, pp. 782-787. IEEE, 1983.

Butler J and Tomizuka M. Trajectory planning for high speed multiple axis contouring systems. In *American Control Conference, 1989*, pp. 87-94.

Chen Y and Desrochers AA. Structure of minimum-time control law for robotic manipulators with constrained paths. In *Robotics and Automation, 1989. Proceedings, 1989 IEEE International Conference on*, pp. 971-976. IEEE, 1989.

Costantinescu D and Croft EA. Smooth and time-optimal trajectory planning for industrial manipulators along specified paths. *Journal of robotic systems* 2000; 17(5): 233-249.

Dong J and Stori JA. A generalized time-optimal bidirectional scan algorithm for constrained feed-rate optimization. *Journal of dynamic systems, measurement, and control* 2006; 128(2): 379-390.

Dong J and Stori JA. Optimal feed-rate scheduling for high-speed contouring. *ASME Journal of manufacturing science and engineering* 2007; 129 (1): 63-76.

Dong J, Ferreira PM, and Stori JA. Feed-rate optimization with jerk constraints for generating minimum-time trajectories. *International Journal of Machine Tools and Manufacture* 2007; 47(12): 1941-1955.

Farouki RT, Tsai YT, and Wilson CS. Physical constraints on feedrates and feed accelerations along curved tool paths. *Computer Aided Geometric Design* 2000; 17(4): 337-359.

Gourdeau R and Schwartz HM. Optimal control of a robot manipulator using a weighted time-energy cost function. In *Decision and Control, 1989, Proceedings of the 28th IEEE Conference on*, pp. 1628-1631. IEEE, 1989.

Imamura F and Kaufman H. Feedrate optimization for machine tool control subject to contour error constraints. In *American Control Conference, 1989*, pp. 81-86.

Imamura F and Kaufman H. Time optimal contour tracking for machine tool controllers. *IEEE Transactions on Control Systems,* 1991; 11(3): 11-17.

Mattmüller J and Gisler D. Calculating a near time-optimal jerk-constrained trajectory along a specified smooth path. *The International Journal of Advanced Manufacturing Technology* 45, no. 9-10 (2009): 1007-1016.

Polit S and Dong J. Development of a high-bandwidth XY nanopositioning stage for high-rate micro-/nanomanufacturing. *IEEE/ASME Transactions on Mechatronics* 2011; 16 (4): 724-733.

Renton D and Elbestawi MA. High speed servo control of multi-axis machine tools. *International Journal of Machine Tools and Manufacture* 2000; 40(4): 539-559.

Shiller Z. On singular time-optimal control along specified paths. *IEEE Transactions on Robotics and Automation* 1994; 10(4): 561-566.

Shin KG and McKay N. Minimum-time trajectory planning for industrial robots with general torque constraints. In *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol. 3, pp. 412-417. IEEE, 1986.

Tarkiainen M and Shiller Z. Time optimal motions of manipulators with actuator dynamics. In *Proceedings of 1993 IEEE International Conference on Robotics and Automation, 1993*, pp. 725-730.

Timar SD, Farouki RT, Smith TS, and Boyadjieff CL. Algorithms for time–optimal control of CNC machines along curved tool paths. *Robotics and Computer-Integrated Manufacturing* 2005; 21(1): 37-53.

Zlajpah L. On time optimal path control of manipulators with bounded joint velocities and torques. In *Robotics and Automation, 1996. Proceedings, 1996 IEEE International Conference on*, vol. 2, pp. 1572-1577. IEEE, 1996.