Paper
Int'l J. of Aeronautical & Space Sci. 15(2), 183–189 (2014)
DOI:10.5139/IJASS.2014.15.2.183

Generalized Computational Nodes for Pseudospectral Methods

Chang-Joo Kim*, Soo Hyung Park, Sung-Nam Jung*** and Sangkyung Sung******

Konkuk University, Seoul, Korea, 143-701

Abstract

Pseudo-spectral method typically converges at an exponential rate. However, it requires a special set of fixed collocation points (CPs) to get highly accurate formulas for partial integration and differentiation. In this study, computational nodes for defining the discrete variables of states and controls are built independently of the CPs. The state and control variables at each CP, which are required to transcribe an NOCP into the corresponding NLP, are interpolated, using those variables allocated at each node. Additionally, Lagrange interpolation and spline interpolation are investigated, to provide a guideline for selecting a favorable interpolation method. The proposed techniques are applied to the solution of an NOCP using equally spaced nodes, and the computed results are compared to those using the standard PS approach, to validate the usefulness of the proposed methods.

Key words: Pseudo-spectral method, computational node, Lagrange interpolation, spline

1. Introduction

The pseudo-spectral (PS) method has been widely used in analyses of nonlinear optimal control problems (NOCPs). This method converts the NOCP into a nonlinear programming problem (NLP), using Lagrange interpolating polynomials and a quadrature formula. It is superior to other direct methods, in that it requires relatively few computational nodes (CNs), compared to other strategies, and the resultant solution can accurately approximate the optimality condition of Euler-Lagrange equations for the NOCP [1-3]. The detail algorithms used in this study are described in the authors' earlier works [4-5], with their applications to rotorcraft flight dynamic analyses. For problems with a well-behaved solution, the PS method typically converges at an exponential rate [1-2, 6-7]. However, it requires a special set of fixed collocation points (CPs) to get highly accurate formulas for partial integration and differentiation. This may cause inconvenience in many applications. First of all, when the end point is not included in the CPs, boundary controls cannot be predicted. Also, relatively small values of the quadrature weights around two end points can reduce the sensitivity of an NLP to the parameterized control variables. As a result, control predictions show poor convergence around these points. Furthermore, when the optimum solution changes rapidly in the middle of the time domain, an extremely large number of CNs is required to provide a discernible improvement in local accuracy. To mitigate these disadvantages, this paper studies generalized CNs for the PS method.

In this study, CNs, on which the discrete variables of states and controls are defined, are built independently of the CPs. The state and control variables at each CP, which are required to transcribe an NOCP into the corresponding NLP, are interpolated, using those variables allocated at CNs. Since the resultant NLP is parameterized using states and controls defined in the CN, its sensitivity to the design variables can be modified through the interpolation process. Therefore, the success of this approach depends on a careful selection of an interpolation method. In this regard, Lagrange interpolation and spline interpolation are investigated, to provide a guideline for selecting a favorable interpolation method. The proposed techniques are applied to the solution of an NOCP using equally spaced CNs, and the computed results are compared to those using the standard PS approach, to

* Associate Professor
** Associate Professor
*** Professor
**** Associate Professor, Corresponding Author : sksung@konkuk.ac.kr

Received: March 6, 2014 Revised : June 5, 2014 Accepted: June 9, 2014

validate the usefulness of the proposed methods.

2. Generalization of Computational Nodes

The PS method typically utilizes a specially designed CN that can be represented by the union of the (K+1) CP $\{\tilde{\tau}_k\}_{k=0}^{k=K}$ and the end points {-1, 1} [4, 6-7]. Fig. 1 shows examples of a generalized CN (GCN), when twenty CPs are used, with *K*=19. The Legendre-Gauss (LG) CN corresponds exactly to the LG CP, except for the two end points that are not included in the LG CP. Three different examples of GCNs are illustrated, two of which are locally clustered, to enhance local accuracy. The third one is built with uniformly distributed nodes, which are known to be the worst possible choice when polynomial interpolation is used, as in the PS method [8, 9]. An NOCP suitable for solution using the PS method can be represented by Eqs. (1) – (3), after applying the affine transformation defined in Eq. (4), over a time horizon of $t \in [t_0, t_f]$ [10-11].

$$\min_{x,u,t_f} J(\mathbf{x},\mathbf{u},t_f) = \varphi(\mathbf{x}(t_0),\mathbf{x}(t_f)) + \frac{t_f - t_0}{2}\int_{-1}^{1} F(\mathbf{x},\mathbf{u},\tau)d\tau \qquad (1)$$

subject to:

$$\frac{d\mathbf{x}}{d\tau} = \frac{t_f - t_0}{2}\mathbf{f}(\mathbf{x},\mathbf{u},\tau) \qquad (2)$$

$$\begin{aligned} &\mathbf{h}(\mathbf{x},\mathbf{u},\tau) = 0 \\ &\mathbf{g}(\mathbf{x},\mathbf{u},\tau) \le 0 \\ &\boldsymbol{\psi}(\mathbf{x}(t_0),\mathbf{x}(t_f)) = 0 \end{aligned} \qquad (3)$$

$$\tau = 2\frac{t - t_0}{t_f - t_0} - 1 \qquad (4)$$

The PS method transcribes Eq. (1) using quadrature weights $\{w_k\}_{k=0}^{k=K}$, as represented by Eq. (5), and the system dynamics can be parameterized using either the differentiation method or the integration method, the transcribed results of which are shown in Eq. (6) and Eq. (7), respectively, for *j*=0, ..., *K*.

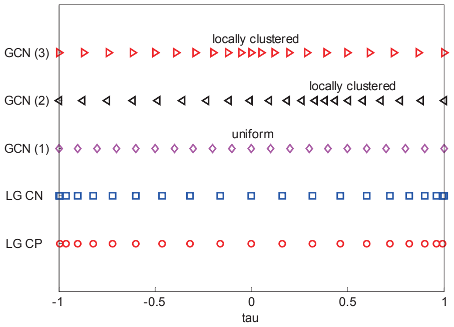


Fig. 1. Examples of generalized computational nodes (M=21)

$$J \approx \varphi\big(\mathbf{x}(t_0),\mathbf{x}(t_f)\big) + \frac{t_N - t_0}{2}\sum_{k=0}^{K} w_k F(\tilde{\mathbf{x}}_k,\tilde{\mathbf{u}}_k,\tilde{\tau}_k) \qquad (5)$$

$$\sum_{k=0}^{K} D_{jk}\tilde{\mathbf{x}}_k = D_j^{(0)}\mathbf{x}(t_0) + \frac{t_f - t_0}{2}\mathbf{f}\big(\tilde{\mathbf{x}}_k,\tilde{\mathbf{u}}_k,\tilde{\tau}_k\big) \qquad (6)$$

$$\tilde{\mathbf{x}}(\tilde{\tau}_j) \approx \mathbf{x}(t_0) + \frac{t_f - t_0}{2}\sum_{k=0}^{K} I_{jk}\mathbf{f}\big(\tilde{\mathbf{x}}_k,\tilde{\mathbf{u}}_k,\tilde{\tau}_k\big) \qquad (7)$$

where, $\tilde{\mathbf{x}}_k$ and $\tilde{\mathbf{u}}_k$ represent the state and control variables corresponding to the k^{th} CP. The terms D_{jk} and I_{jk} are the elements of the differentiation and integration matrices. $D_j^{(0)}$ reflects the effect of initial states when the initial time is not included in the CPs, as in the LG, and the flipped Legendre-Gauss-Radau (FLGR) points [12]. By satisfying the equality and inequality constraints shown in Eq. (3) at each of the CPs and the two end points, the NOCP is completely transformed into a solvable NLP.

For convenience, we can differentiate the state and control variables at each of the CPs, $\{\tilde{\mathbf{x}}_k,\tilde{\mathbf{u}}_k\}_{k=0}^{k=K}$, from those corresponding to the CNs, $\{\mathbf{x}_m,\mathbf{u}_m\}_{n=0}^{m=M}$, where *M* is typically one of {*K*, *K*+1, *K*+2}, depending on whether or not the end points are incorporated in the CPs. For the application of the GCN, the following interpolation formula is defined to approximate the continuous states in the computational domain, using states $\{\mathbf{x}_m\}_{m=0}^{m=M}$ at each of the GCN.

$$\mathbf{x}(\tau) = \sum_{l=0}^{M}\phi_l(\tau)\mathbf{x}_l \ \text{ with } \ \phi_l(\tau_m) = \delta_{lm} \qquad (8)$$

Then, the states corresponding to the kth point of the CP can be estimated, using:

$$\tilde{\mathbf{x}}_k = \sum_{l=0}^{M}\tilde{\phi}_{kl}\mathbf{x}_l \ \text{ with } \ \tilde{\phi}_{kl} = \phi_l(\tilde{\tau}_k) \qquad (9)$$

The coefficient $\tilde{\phi}_{kl}$ is an influence coefficient with respect to $\mathbf{x}_l$, and the same formula can be used to approximate the control $\tilde{\mathbf{u}}_k$, using $\{\mathbf{u}_l\}_{l=0}^{l=M}$. Therefore, the solution of the NLP represented with the variables at each of the CP and the two end points can be carried out with the following procedures:

(i) Compute states and controls $\{\tilde{\mathbf{x}}_k,\tilde{\mathbf{u}}_k\}_{k=0}^{k=K}$ at each of the CPs, using Eq. (8).

(ii) Compute the cost and constraint functions that are parameterized, using the variables at each of the CPs, and the two end points.

(iii) Compute the Jacobian matrices for the cost and constraint functions, with respect to $\{\mathbf{x}_m,\mathbf{u}_m\}_{m=0}^{m=M}$.

(iv) Solve the NLP using the robust SQP (rSQP) algorithm, and update $\{\mathbf{x}_m,\mathbf{u}_m\}_{m=0}^{m=M}$.

(v) Terminate the routine, if the solution converges.

Otherwise, repeat steps (i) through (iv).

The Jacobians in step (iii) can be easily approximated, using the influence coefficient matrix. As an example, the gradient of the integral cost function, as shown in Eq. (5), with respect to a control vector $\mathbf{u}_m$, can be estimated using Eq. (11).

$$J_I \approx \sum_{k=0}^{K} w_k F(\tilde{\mathbf{x}}_k, \tilde{\mathbf{u}}_k, \tilde{\tau}_k) \tag{10}$$

$$\begin{aligned} \frac{\partial J_I}{\partial \mathbf{u}_m} &\approx \sum_{k=0}^{K} w_k \frac{\partial F(\tilde{\mathbf{x}}_k, \tilde{\mathbf{u}}_k, \tilde{\tau}_k)}{\partial \tilde{\mathbf{u}}_k} \frac{\partial \tilde{\mathbf{u}}_k}{\partial \mathbf{u}_m} \\ &= \sum_{k=0}^{K} \tilde{w}_{km} \frac{\partial F(\tilde{\mathbf{x}}_k, \tilde{\mathbf{u}}_k, \tilde{\tau}_k)}{\partial \tilde{\mathbf{u}}_k} \end{aligned} \quad \text{with} \quad \tilde{w}_{km} = w_k \tilde{\phi}_{km} \tag{11}$$

The state and control variables $\{\mathbf{x}_m, \mathbf{u}_m\}_{m=0}^{m=M}$ defined at the GCN become the design variables in the resultant NLP. The modified weight $\tilde{w}_{km}$ in Eq. (11) reflects the sensitivity of the NLP to $\{\mathbf{x}_m, \mathbf{u}_m\}$, which depends on both the type of the interpolation function $\phi_l(\tau)$, and the distribution of the GCN, as shown in Fig. 1. Therefore, a careful selection of interpolation functions, as well as types of GCN, is crucial for the successful implementation of the GCN, by preserving sufficient sensitivity to all of the design variables.

This paper considers two different types of interpolation. Because the PS method utilizes the Lagrange interpolating polynomials to approximate states and controls in a continuous time domain, they seem to be a natural choice for accurate interpolation for Eq. (8). However, the computation of the influence coefficient matrix using those functions needs a matrix inversion, which may degrade interpolation accuracy, as a result of the large condition number of the related matrix, when the number of CPs is increased. As an alternative, the local Lagrange interpolating polynomial is tested as the first type of interpolation in this study, and the spline interpolation is considered as the second approach. For this purpose, some manipulations are required to obtain a formula similar to Eq. (8), using spline interpolation polynomials. In case an N^{th}-order polynomial is used, a continuous function *f(t)* can be approximated using the discrete data set $\{f_m, t_m\}_{m=0}^{M}$, and the related spline interpolation can be represented by:

$$f(t) \approx \tilde{f}_m(\eta_m) = \sum_{n=0}^{N} a_{m,n} \eta_m^n \ , \ t \in [t_m, t_{m+1}] \tag{12}$$

where, the non-dimensional time η_m is defined as:

$$\eta_m = \frac{t - t_m}{t_{m+1} - t_m}, \quad \eta_m \in [0,1] \tag{13}$$

The unknown coefficient set $\{a_{m,n}\}_{m=0,n=0}^{m=M,n=N}$ can be determined by imposing various conditions, as listed below.

(i) Function values given at each point

(ii) Continuity of the function and its derivatives up to the (N-1)th-order, at each of the inter-connecting points

(iii) (N-1) end conditions

There are many different ways of imposing the end conditions, but this paper limits its consideration to the natural spline and the not-a-knot spline [13-15]. The conditions above produce a system of linear algebraic equations for the unknown coefficients, as represented in Eq. (14):

$$\mathbf{Xa} = \mathbf{Cf} \tag{14}$$

where, **a** and **f** are defined by:

$$\begin{aligned} \mathbf{a} \ &= (\mathbf{a}_0^T, \mathbf{a}_1^T, \cdots, \mathbf{a}_{M-1}^T)^T \\ \mathbf{a}_m &= (a_{m,0}, a_{m,1}, \cdots, a_{m,N})^T, \quad m = 0,1,\cdots, M-1 \\ \mathbf{f} \ &= (f_0, f_1, \cdots, f_M)^T \end{aligned}$$

The coefficient matrix **c** has the dimensions M(N+1)-by-(M+1), and is defined to show that each unknown coefficient can be represented as a linear combination of function values $\{f_k\}_{k=0}^{M}$, as shown below:

$$a_{m,n} = \sum_{k=0}^{M} \alpha_{m,n,k} f_k \tag{15}$$

The coefficient set $\{\alpha_{m,n,k}\}_{m=0,n=0}^{m=M,n=N}$, which is related to the k^{th} function data f_k, can be obtained by solving Eq. (14), after setting $f_j = \delta_{jk} \ \ (j = 0,\cdots, M)$. Therefore, an approximation of the function can be rewritten using these coefficients, as:

$$f(t) \approx \tilde{f}_m(\eta_m) = \sum_{k=0}^{M} \tilde{\phi}_{m,k}(\eta_m) f_k \tag{16}$$

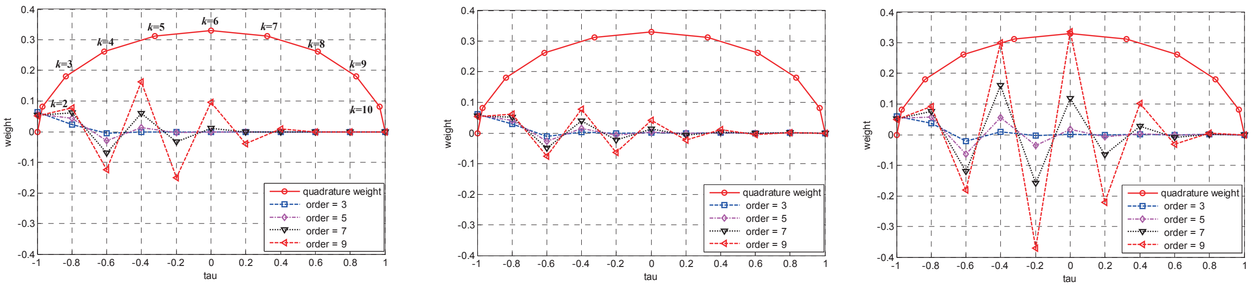
where, the interpolation polynomial is expressed by:

$$\tilde{\phi}_{m,k}(\eta_m) = \sum_{n=0}^{N} \alpha_{m,n,k} \eta_m^n \tag{17}$$

Since Eq. (16) has the same form as Eq. (8), the corresponding influence coefficient matrix can be easily computed.

3. Applications and Discussions

As previously mentioned, a GCN can be built adaptively for the analysis purposes, to obtain a highly flexible node strategy. However, a GCN generated with an equal step size is utilized in this paper, to validate simply the usefulness of the proposed method. Fig. 2 shows how the modified weights can vary with 11 nodes, depending on the types of interpolation. The quadrature weight w_2 corresponding to the second CP is distributed over many of the CNs; and the sensitivity of the NLP to the initial state and control variables is restored with $w_2\widetilde{\phi}_{20} \neq 0$. Therefore, the boundary controls can be predicted, even using the LG CP, without resorting to applying either an inaccurate extrapolation, or a rigorous approach using Pontryagin's minimum principle, as used in Ref. [7]. As the order of the interpolation functions is increased, each quadrature weight spreads its influence more strongly to many of the GCNs. However, the highly oscillatory behavior with higher-order polynomials may degrade solution accuracy, and cause numerical divergence. As shown in Fig. 2, the natural spline seems to be the best among the tested interpolation methods, in that it generates a much more even distribution of the modified weights, than other approaches. Therefore, follow-on applications of the GCN are limited to the use of the natural spline. Fig. 3 represents the modified quadrature weights with the septic natural spline.



(a) Local Lagrange interpolation

(b) Natural spline

(c) Not-a-knot spline

Fig. 2. Modified weights depending on interpolation functions with equally spaced computational nodes (*k*=2, *M*=11)

For applications of the proposed GCN approach to the solution of an NOCP, the following minimum energy problem with a path constraint, which has an analytic solution, is selected [10].

$$\text{Min } J = \int_0^1 u^2 dt$$

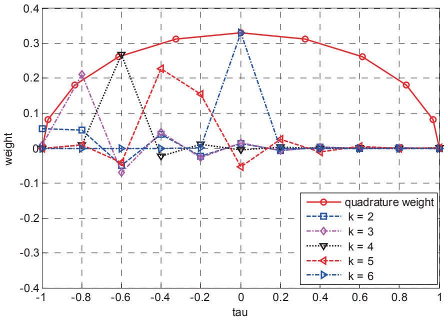


Fig. 3. Modified weight distribution with the septic natural spline (7th order)

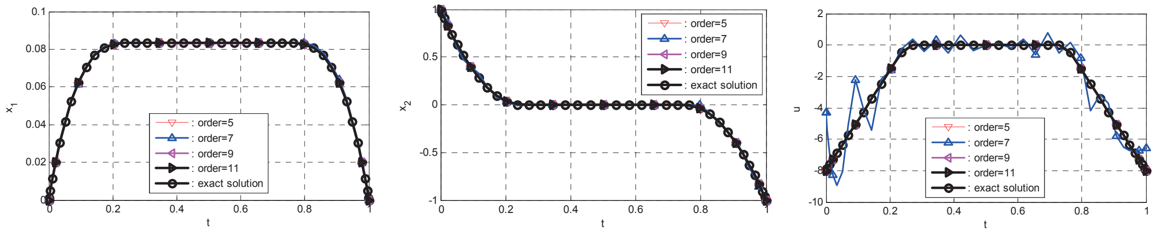


Fig. 4. Analysis results at the collocation points(natural spline, 41 computational nodes)

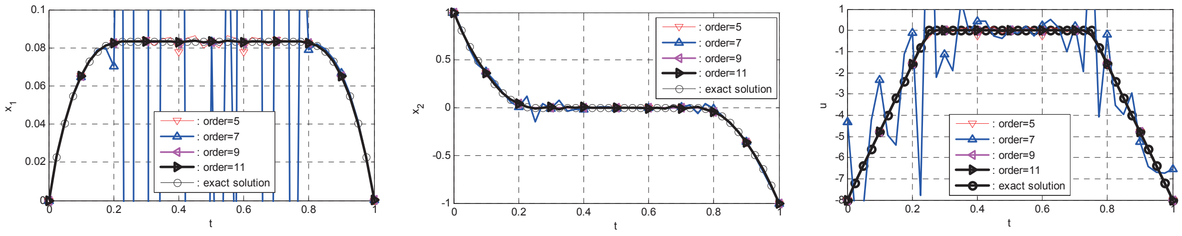


Fig. 5. Analysis results at the generalized computational nodes(natural spline, 41 computational nodes)

subject to:

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix}=\begin{pmatrix}x_2\\ u\end{pmatrix},\quad \begin{matrix} x_1(0)=0,\ x_1(1)=0\\ x_2(0)=1,\ x_2(1)=-1\end{matrix}$$

$$x_1(t)\le 1/12$$

The NOCP analysis is initialized with $x_1(t)$=$x_2(t)$=0 and $u(t)$=-4 for all CNs. The analysis results with varying orders of the interpolation polynomials are compared in Figs. 4-6. The solution with the septic spline (N=7) tends to be divergent, and the quantic spline (N=5) presents a relatively large difference between two solutions, corresponding to the CP and GCN, respectively. This difference can be used as a measure of inaccuracy of the applied interpolation formula. Fig. 6 shows the convergence of the cost function prediction. The PS method using the GCN and the 9th- or 11th-order natural spline interpolations outperforms the standard method, in that it presents faster initial convergence. Fig. 7 supports this claim, by showing that the solution based on the GCN provides better initial convergence in control prediction around two end points, with enhanced sensitivity to the corresponding controls.

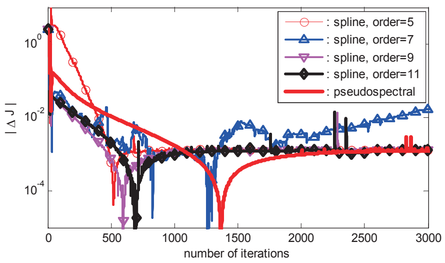
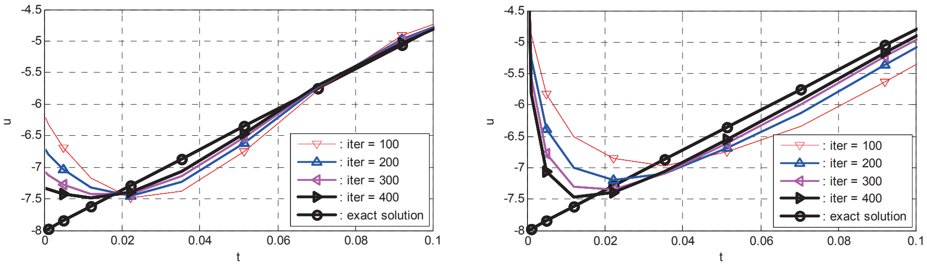


Fig. 6. Convergence history of the cost function error(natural spline interpolation, nodes = 41)

One major drawback in using the GCN is related to the structure of the corresponding NLP. The reformulated NLP with the GCN generates a denser structure in the Karush-Kuhn-Tucker system, than that with the standard PS approach. Further, the computation of the influence coefficients may become inaccurate, as the number of nodes is increased. These problems can be mitigated through the use of knotting techniques [16]. Figs. 8-10 show the results of applying a knotting technique to the GCN. The converged solution with the standard PS method using LG CP cannot predict the controls at two end points, and at the knot located at t=0.6. On the other hand, the solution with the GCN provides better convergence in control prediction around these points, than that using the standard PS method. Fig. 11 shows the application of the GCN with the differentiation and integration methods, the results of which prove that there are no limits to the application of the proposed GCN method to the PS method.



(a) Generalized CN

(b) Pseudospectral CN

Fig. 7. Convergence history of the control prediction(ninth-order natural spline, nodes = 41)

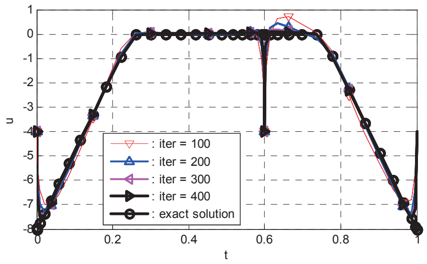
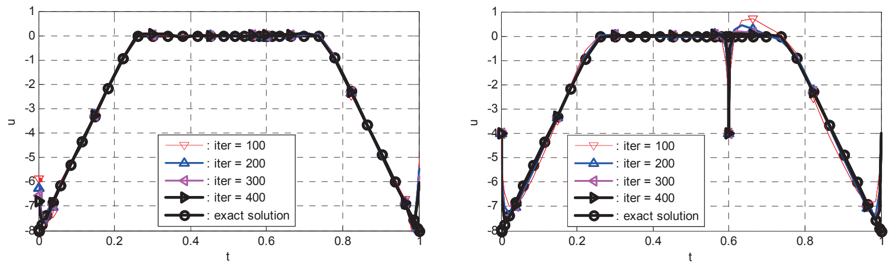


Fig. 8. Comparison of control predictions, after 5000 iterations with the knotting technique (ninth-order natural spline interpolation, nodes = 41)

4. Conclusions

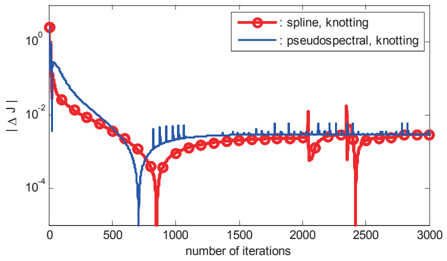
This paper proposed the application of generalized computational nodes, to reduce inconveniences in solving a nonlinear optimal control problem with the standard pseudospectral method. For the interpolation of the state and control variables at each of the collocation points, Lagrange interpolating polynomials and spline functions were tested, to select the most favorable interpolation



(a) Generalized CN

(b) Pseudospectral CN

Fig. 9. Convergence characteristics of control predictions with the knotting technique(ninth-order natural spline interpolation, nodes = 41)



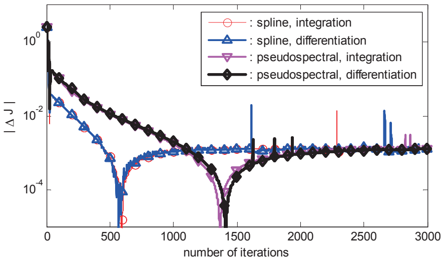


Fig. 10. Convergence of the cost functions with the knotting technique(ninth-order natural spline interpolation, nodes = 41)

Fig. 11. Comparison of the transcription method(ninth-order natural spline interpolation, nodes = 41)

method. The natural spline functions outperformed the other interpolation functions tested, with considerably fewer oscillatory behaviors in the modified quadrature weights. The state and control variables predicted at the generalized computational nodes can show large differences from those at the collocation points. Since this difference can be a measure of inaccurate interpolation, it should be minimized, by using a suitably chosen leading order of the natural spline functions. The application of generalized computational nodes showed faster initial convergence in control prediction, than the standard pseudospectral approach. In addition, the knotting technique could be applied to enhance sparsity in the Karush-Kuhn-Tucker system, when generalized computational nodes are used.

Acknowledgments

This paper was written as part of Konkuk University's research support program for its faculty on sabbatical leave, in 2013.

References

[1] Williams, P., "A Gauss-Lobatto Quadrature Method for Solving Optimal Control Problems," *Australian and New Zealand Industrial and Applied Mathematics Journal*, Vol. 47, 2005, pp. 101-115.

[2] Fahroo, F. and Ross, I. M., "Costate Estimation by a Legendre Pseudo-spectral Method," *AIAA Journal of Guidance, Control and Dynamics*, Vol. 24, No. 2, 2001, pp. 270-277.

DOI: 10.2514/2.4709

[3] Gong, Q., Ross, I. M., Kang, W., and Fahroo, F., "Connections between the Covector Mapping Theorem and Convergence of Pseudo-spectral Methods for Optimal Control," *Computational Optimization and Applications*, Vol. 41, No. 3, 2008, pp. 307-335.

[4] Kim, C.-J., Sung, S., and Shin, K., "Pseudo-spectral Application to Nonlinear Optimal Trajectory Generation of a Rotorcraft," *The First International Conference on Engineering and Technology Innovation*, Kenting, Taiwan, November 11-15, 2011.

[5] Kim, C.-J., Sung, S., Park, S. H., and Jung, S. N., " Time-Scale Separation for Rotorcraft Nonlinear Optimal Control Analyses, "*Journal of Guidance, Control and Dynamics*, Vol. 37, No. 2, March–April 2014, pp. 655–673.

[6] D. Benson, "A Gauss Pseudo-spectral Transcription for Optimal Control," MIT, Ph.D. Thesis, Department of Aeronautics and Astronautics, November 2004.

[7] G. T. Huntington, "Advancement and Analysis of a Gauss Pseudo-spectral Transcription for Optimal Control Problems," Ph. D. Thesis, MIT, June 2007.

[8] I. M. Ross and M. Karpenko, "A Review of Pseudospectral Optimal Control: From theory to flight," *Annual Reviews in Control*, Vol. 36, No. 2, December 2012, pp. 182–197.

[9] Q. Gong, I. M. Ross, and F. Fahroo, "Pseudospectral Optimal Control on Arbitrary Grids," *AAS/AIAA Astrodynamics Specialist Conference*, Pittsburgh, PA, August 9-13, 2009.

[10] Bryson, A. E., Jr., and Ho, Y. C., *Applied Optimal Control*, Hemisphere Publishing, Washington D.C., 1975.

[11] Kirk, D. E., *Optimal Control Theory*; An Introduction, Dover, New York, 1970.

[12] Garg, D., Patterson, M., Hager, W. W., Rao, A. V., Benson, D. A. and Huntington, G. T., "A Unified Framework for the Numerical Solution of Optimal Control Problems Using Pseudospectral methods," *Automatica*, Vol. 46, No. 11, November 2010, pp. 1843–1851.

DOI:10.1016/j.automatica.2010.06.048

[13] Vijaya Bhaskar, N. R. Babu, and K. Varghese, "Spline Based Trajectory Planning for Cooperative Crane Lifts," *Proceedings of the 23rd ISARC*, Tokyo, 2006, pp. 418-423

[14] Ogundare, B. S. and Okecha, G. E., "A Pseudo Spline Methods for Solving an Initial Value Problem of Ordinary Differential Equation," *Journal of Mathematics and Statistics*, Vol. 4, No. 2, 2008, pp. 117-121

[15] Akram, G. and Siddiqi, S. S., "End conditions for interpolatory septic splines," *International Journal of Computer Mathematics*, Vol. 82, No. 12, December 2005, pp. 1525–1540.

[16] Ross, I. M. and Fahroo, F., "Pseudo-spectral Knotting Methods for Solving Optimal Control Problems," *Journal of Guidance, Control and Dynamics*, Vol. 27, No. 3, May–June 2004, pp. 397–405.

DOI: 10.1109/TAC.2005.860248