

Incremental Predictive Command of Velocity to Be Gained Guidance Method

N. Ghahramani¹, A. Naghash², F. Towhidkhal³

In this paper, a new incremental predictive guidance method based on implicit form of velocity to be gained algorithm is proposed. In this approach, the generalized incremental predictive control (GIPC) approach is applied to the linearized model to compensate for the guidance error. Instead of using the present state in popular model based predictive controller (MPC), in the new method both previous and present states are utilized. GIPC approach introduces a feedback action including the weighted difference of the process states and the summation of the control action increments.

To evaluate the robustness and performance of the proposed approach, the parameter uncertainties of the guidance and control are considered and a comparison with standard GPC is performed by extensive computer simulations. The results show a significant improvement in the robustness as well as tracking performance of the perturbed initial value of velocity to be gained or the reference signal.

INTRODUCTION

The inertial closed loop guidance schemes can be classified into two categories, namely, path adaptive schemes and perturbation schemes [1]. In the first type, the steering command is generated from the solution of the simplified equations of motion. The second type assumes that the launch trajectory is defined completely before the launch and the problem is to find the optimum steering logic that forces the vehicle to follow the nominal trajectory. This type is simple and the accuracy achieved is high [2]. The Q-guidance [3, 4] is prominent among the perturbation guidance schemes. The method uses the velocity to be gained concept, and tries to drive it to zero.

Recently, ideas taken from linear quadratic (LQ) theories have been applied to guidance problem [5, 6, 7], and less attention is paid to the model based predictive control (MPC) schemes in this application. MPC essentially solves standard optimal control prob-

lems using a finite horizon in contrast to the infinite horizon required in the LQ based algorithms. This provides a practical approach to complex systems with uncertainties in which LQ algorithms encounter difficulties in optimization procedure. Moreover, MPC benefits from on-line computations for the current state of the plant, but the computations in LQ algorithm is off-line based on a feedback policy that provides the optimal control for all states.

A variety of predictive controllers and the history of them have been explained in the literature [8-11]. Among these, generalized predictive control (GPC) [12, 13] has become one of the most popular MPC methods both in industry and academia. Although T filter has been suggested to improve the robustness of GPC [14, 15], this filter needs a redesign of the controller. Poor performance or robustness of an MPC controller is often due to poor modeling assumptions [16]. For this reason, modeling is the most important part of an MPC design. Various predictive control models (state space and input-output) have been employed by different MPC algorithms. One of them is the incremental form for the control action [10, 17, 18]. In these approaches, the state increments or the differences of states have not been included.

This paper presents a new incremental predictive scheme of Q-guidance in order to compensate for the

-
1. Assistant Professor, Malek Ashtar Univ., Tehran, Iran, Formerly Ph.D. Candidate, Dept. of Aerospace Eng., Amirkabir Univ. of Tech., Tehran, Iran.
 2. Assistant Professor, Dept. of Aerospace Eng., Amirkabir Univ. of Tech., Tehran, Iran.
 3. Associate Professor, Dept. of Biomedical Eng., Amirkabir Univ. of Tech., Tehran, Iran.

known perturbed guidance errors with emphasis on robust performance. In aforementioned MPC approaches, only present states are used in states prediction. But, in this new receding horizon algorithm, both present and previous states are considered. For this reason, the proposed guidance command includes the weighted difference of the current and the previous states and the integration of the command action increments. Using the weighted difference of states in guidance command improves the algorithm robustness with respect to the guidance and control system uncertainties. The special feature of the proposed predictive guidance approach is to combine the current and future guidance errors, while obtaining optimal command and desired robust performance. Since this new method can compensate for the predicted future guidance error (in this paper it is named the perturbed guidance error), it increases the accuracy of the standard Q-guidance method especially for missiles with large trajectory deviations. The effectiveness of the proposed method is verified through extensive simulations in the presence of the guidance and control model uncertainties.

PREDICTIVE GUIDANCE SCHEME

The main task of a guidance algorithm is to obtain a command that is a function of the guidance error. The error is the difference between the reference trajectory $w(k)$, and the guidance output signal. Here, the compensation for the guidance error for the known reference trajectory using incremental predictive control is considered, and the predictive guidance problem with the following nonlinear dynamic equation is formulated as follows (see Appendix):

$$x(k+1) = A(k)x(k) + f(u(k)), \quad y(k) = x(k) \quad (1)$$

In this equation, $x \in R$ denotes the state, $u, y \in R$ are the process input and the process output, respectively; $f(\cdot)$ is a differentiable function, and $A(k)$ is a time varying coefficient of the nonlinear dynamic equation.

The guidance methods based on predictive control algorithms is able to obtain the future predicted guidance error over the prediction horizon n_y (i.e., $w(k+j) - y(k+j)$, $j = 1, \dots, n_y$) and then, compute the optimal guidance command over the finite control horizon n_u . The state or output response of the guidance dynamics is predicted by Eq. (1) and a quadratic cost function J based on the predicted errors and the increment of the current guidance command $\Delta u(\cdot)$ is minimized:

$$J = \sum_{j=1}^{n_y} [w(k+j) - y(k+j)]^2 + \sum_{j=1}^{n_u} r(j) \Delta u(k+j-1)^2 \quad (2)$$

in which $r(\cdot)$ is a control weighting sequence. The optimization of this cost function is subject to Eq. (1) as well as the constraint:

$$\Delta u(k+i) = 0, \quad i = n_u, \dots, n_y \quad (3)$$

The minimization produces the sequences $\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+n_u-1)$, but only $\Delta u(k)$ is actually applied. At time $k+1$, a new minimization problem is solved. This implementation is called receding horizon guidance, which significantly improves the performance of the guidance system.

INCREMENTAL PREDICTIVE COMMAND DESIGN

Various predictive control models (state space and input-output) have been employed by different MPC algorithms. One of them is the incremental form for the control action [17, 18]. In spite of the incremental methods suggested previously, the method proposed in this paper is based on the state increments instead of control action (guidance command). To derive the state space incremental form, we assume that the coefficients of Q-guidance in Eq. (1) are constant over the prediction horizon. With this assumption, and by using on-line linearization [19, 20], the incremental form of the dynamical Eq. (1) is given by:

$$\Delta x(k+1) = A\Delta x(k) + \left. \frac{\partial f}{\partial u} \right|_{u=u^*} \Delta u(k) = A\Delta x(k) + B\Delta u(k) \quad (4)$$

where $B = \left. \frac{\partial f}{\partial u} \right|_{u=u^*}$ and

$$\begin{aligned} \Delta u(k) &= u(k) - u(k-1), \\ \Delta x(k) &= x(k) - x(k-1), \\ \Delta x(k+1) &= x(k+1) - x(k) \end{aligned} \quad (5)$$

Combining Eqs. (4) and (5) gives:

$$x(k+1) = (A+1)x(k) - Ax(k-1) + B\Delta u(k) \quad (6)$$

After some algebraic manipulations, the general form of Eq. (6) is given by:

$$y(k+j) = \left(\sum_{m=0}^j A^m \right) x(k) - \left(\sum_{m=1}^j A^m \right) x(k-1) + \sum_{n=1}^j \left(\sum_{m=0}^{n-1} A^m \right) B \Delta u(k+j-n) \quad (7)$$

Compared to the standard state space form, this new form described by Eq. (7) has a new term (second term), which comes from the incremental form of

states. Using Eq. (7), the incremental form of state space model can be written as the following compact matrix/vector form:

$$\underline{\mathbf{y}}_k = \mathbf{P}_{y0}x(k) - \mathbf{P}_{y1}x(k-1) + \mathbf{H}_y\Delta\underline{\mathbf{u}}_{k-1} \quad (8)$$

where:

$$\underline{\mathbf{y}}_k = \begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \\ \vdots \\ y(k+n_y) \end{bmatrix}, \quad \Delta\underline{\mathbf{u}}_{k-1} = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+n_u-1) \end{bmatrix}$$

$$\mathbf{P}_{y0} = \begin{bmatrix} A+1 \\ A^2+A+1 \\ A^3+A^2+A+1 \\ \vdots \\ \sum_{i=0}^{n_y} A^i \end{bmatrix} \in R^{n_y}$$

$$\mathbf{P}_{y1} = \begin{bmatrix} A \\ A^2+A \\ A^3+A^2+A \\ \vdots \\ \sum_{i=1}^{n_y} A^i \end{bmatrix} \in R^{n_y},$$

$$\mathbf{H}_y = \begin{bmatrix} B & 0 & \dots & 0 \\ (A+1)B & B & 0 & \dots & 0 \\ (A^2+A+1)B & (A+1)B & B & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (\sum_{i=0}^{n_y-1} A^i)B & (\sum_{i=0}^{n_y-2} A^i)B & \dots & B \end{bmatrix} \in R^{n_y \times n_u} \quad (9)$$

The predictor guidance law is determined by minimizing the deviation of the predicted guidance response from a specified target reference over a prediction horizon as $\underline{\mathbf{w}}_k = [w(k+1), w(k+2), \dots, w(k+n_y)]^T$. The cost function J described by Eq. (2) can be rewritten as:

$$J = (\underline{\mathbf{w}}_k - \underline{\mathbf{y}}_k)^T (\underline{\mathbf{w}}_k - \underline{\mathbf{y}}_k) + \Delta\underline{\mathbf{u}}_{k-1}^T \mathbf{R} \Delta\underline{\mathbf{u}}_{k-1},$$

$$\mathbf{R} = r \mathbf{I} > 0 \quad (10)$$

in which \mathbf{R} is a positive definite matrix (For a symmetric matrix $\mathbf{M} \in R^{n \times n}$ and a vector $\mathbf{x} \in R^n$, If real valued $\mathbf{x}^T \mathbf{M} \mathbf{x}$ is positive definite, we say that the matrix \mathbf{M} is positive definite and write $\mathbf{M} > \mathbf{0}$). Minimizing the cost function J with respect to $\Delta\underline{\mathbf{u}}_{k-1}$, and solving for it gives the guidance command sequence to be applied to the system:

$$\Delta\underline{\mathbf{u}}_{k-1} = \mathbf{K}[\underline{\mathbf{w}}_k - (\mathbf{P}_{y0}x(k) - \mathbf{P}_{y1}x(k-1))] \quad (11)$$

where:

$$\mathbf{K} = (\mathbf{R} + \mathbf{H}_y^T \mathbf{H}_y)^{-1} \mathbf{H}_y^T \quad (12)$$

The minimization produces $\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+n_u+1)$ but only $\Delta u(k)$ is actually applied. On the other hand, we have:

$$u(k) = u(k-1) + \Delta u(k) \quad (13)$$

At time $k+1$, a new minimization problem is solved. The command described by Eqs. (11) and (13) has some interesting and important properties. The command implements the weighted difference between present and previous states, and the summation of incremental command. Therefore, the weighted differentiation of states and the integration of commands are implemented. The first property improves the closed loop stability, and the second one causes the disturbance rejection and tracking error reduction [21].

GUIDANCE AND CONTROL MODEL

Since the guidance command is realized by the control system which tracks the guidance command using fins or thrusters, there always exists a lag between the guidance and control signals. In order to study the effects of this lag, its model is considered in the guidance loop.

Guidance model

We consider the nonlinear dynamic equation of Q-guidance described by Eq. (23). Linearization of the nonlinear equation results in a linear time varying model with coefficients $A_c(t)$ and $B_c(t)$. $A_c(t)$ has very low variations versus time and we can consider it constant. But, $B_c(t)$ has high variations due to thrust and mass flow rate variations. One way to consider these variations is to divide the time varying coefficients into the finite segments and to obtain their mean values in each segment for scheduling the designed parameters. The mean values considered for a segment of the guidance problem are $\bar{A}_c = 0.002$ and $\bar{B}_c = 40$. The discrete form of the model with coefficients \bar{A}_c and \bar{B}_c is as follows:

$$x(k+1) = \bar{A}^* x(k) + \bar{B}^* (1 + \Delta B) u(k) \quad (14)$$

where:

$$\bar{A}^* = 1.0004, \quad \bar{B}^* = 0.97 \quad (15)$$

and ΔB indicates the relative variations of coefficient \bar{B}^* .

Control model

In order to tune the guidance algorithm accurately, we consider the control system dynamics in the guidance

loop. The control system is usually modeled as a first order dynamics [22]:

$$G_c(k) = \frac{1}{\tau s + 1}, \quad \tau = \tau^* + \Delta\tau \quad (16)$$

in which τ is the true control system time constant, τ^* is the nominal control system time constant and $\Delta\tau$ is the uncertainty of τ . Figure 1 shows the block diagram of the closed loop guidance and control system using the proposed GIPC algorithm to compensate for the perturbations of $\delta V_g(k_0)$ or to track the guidance command.

In this figure, a perturbation of $\delta V_g(\cdot)$ is considered at discrete time k_0 . Using the prediction matrices, GIPC algorithm generates the guidance command $\delta\theta_g(k)$ in order to force the perturbation of velocity to be gained to zero.

SIMULATION RESULTS

To evaluate the proposed predictive guidance algorithm, the tracking performance and guidance command in the presence of control system dynamics and thrust parametric uncertainties are studied and compared with GPC method.

Control system dynamics effect

The guidance method can be accurately tuned by using a guidance law that takes in to account the autopilot lag as a first order dynamics. Considering the control system time constant as $\tau^* = 0.4$, the tuning parameters were set for GPC and GIPC methods as follows:

$$S_{GPC} = S_{GIPC} = \{r, n_u, n_y\} = \{100, 3, 15\} \quad (17)$$

To study the effect of control system dynamics on the performance of the overall closed loop guidance and control system, the velocity to be gained is perturbed 1.0 m/s around the nominal value at discrete time k_0 (the initial value of the guidance algorithm difference equation is 1.0 m/s). Figure 2 demonstrates the time history of the perturbed velocity to be gained and command of Q-guidance using GIPC for $\tau = 0.2$

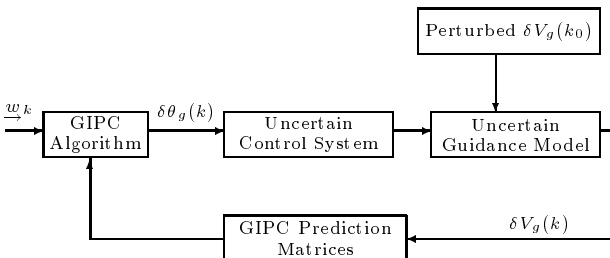


Figure 1. The block diagram of the closed loop guidance and control system using the proposed GIPC algorithm to compensate for the perturbations of $\delta V_g(k_0)$.

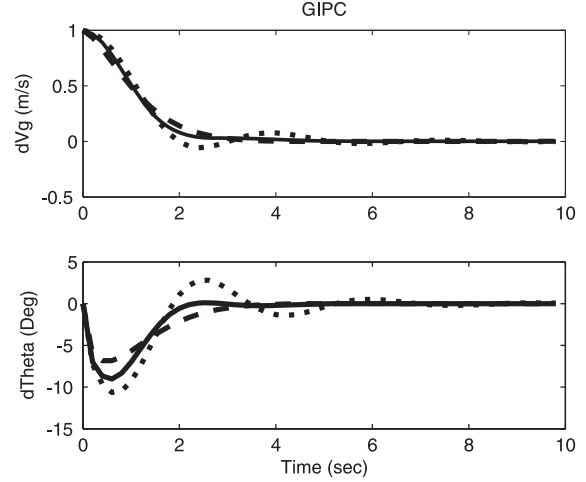


Figure 2. Time history of the state and command of Q-guidance using GIPC for $\tau = 0.2$ (dashed line), $\tau = 0.4$, (solid line) and $\tau = 0.6$ (dotted line).

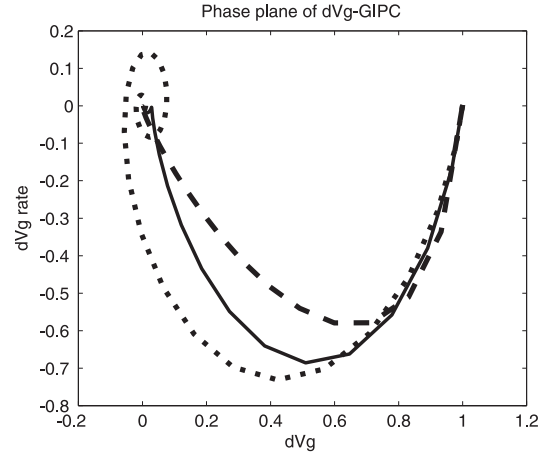


Figure 3. Phase plane of velocity to be gained using GIPC for $\tau = 0.2$ (dashed line), $\tau = 0.4$, (solid line) and $\tau = 0.6$ (dotted line).

(dashed line), $\tau = 0.4$, (solid line) and $\tau = 0.6$ (dotted line). In the nominal case, to compensate for the value of 1.0 m/s for guidance error, the guidance system commands are less than 9 degrees to the control system within 2 seconds. It is observed that the decreasing of τ ($\tau = 0.2$) causes the decreasing of guidance command and the state behavior is approximately as the nominal case ($\tau = 0.4$). When τ increases ($\tau = 0.6$), it affects significantly on the performance of the overall closed loop system (the overshoot and settling time of guidance loop is increased).

Figure 3 illustrates the phase plane of the perturbed velocity to be gained using GIPC for three cases as before ($\tau = 0.2$ (dashed line), $\tau = 0.4$, (solid line) and $\tau = 0.6$ (dotted line)). This figure, interprets Figure 2, and shows that by decreasing τ , the perturbed

velocity to be gained rapidly converges to the origin, and provides a better performance.

Figure 2 and Figure 3 show that for an actual system, in order to improve the guidance performance, the tuning parameters of the algorithm should be accurately tuned in the presence of the control system dynamics.

Robustness with respect to the variations of $B(k)$ and τ

Robustness study of a designed guidance algorithm is a necessary step in design and test procedures for an actual problem. In this section, the effectiveness of the guidance algorithm is demonstrated in the presence of the guidance and control system uncertainties (*i.e.*, ΔB and $\Delta\tau$). For studying the algorithm robustness three cases are considered:

- A) $\Delta B = 0$ and $\Delta\tau = 0.0$ or nominal case (solid line)
- B) $\Delta B = -40\%$, $\Delta\tau = +50\%$ (dashed line)
- C) $\Delta B = +40\%$, $\Delta\tau = +50\%$ (dotted line)

For the above cases, Figure 4a and Figure 4b show the behavior of the state and guidance command using GPC and GIPC approaches, respectively.

It is observed that for the nominal flight condition, two algorithms approximately have the same behaviors, except GIPC has a lower guidance command, and for the deviated conditions (case B and C), GIPC has a smaller settling time and overshoot than GPC. Comparing cases B and C shows that GIPC has a better performance in case C.

Figure 5a and Figure 5b illustrate the phase plane of the deviated velocity to be gained using GPC (dotted line) and GIPC (solid line) methods for Case B and Case C. The initial and final values of the deviated velocity are 0.0 m/s and 1.0 m/s, respectively. These

figures demonstrate that GIPC rapidly converges to the origin and provides a better performance than GPC. The reason for these interesting results is that GIPC employs the weighted difference of the states rather than states to obtain the guidance commands.

CONCLUSIONS

In this paper, an improved generalized predictive control (GPC) algorithm was proposed in order to compensate for the Q-guidance error with the emphasis on robust performance. Instead of using only the present state in popular model based predictive controller, in the new method both previous and present states were utilized. GIPC approach introduces a feedback action including the weighted difference of the process states and the summation of the control action increments.

The proposed approach was evaluated in the presence of model uncertainties of guidance and control system and compared with the standard GPC. It was observed that considering the control system dynamics increases the guidance performance. Moreover, the comparative simulations showed that GIPC is more robust than GPC with respect to the variations of guidance and control modeling parameters. Consequently, from the implementation point of view, the proposed GIPC algorithm is useful to generate the optimal commands of Q-guidance method, especially when the perturbation of velocity to be gained is considered.

APPENDIX: NONLINEAR EQUATION OF GUIDANCE METHOD

It is well known (see References [3, 4, 23]) that the velocity to be gained vector, \mathbf{V}_g , which is defined as

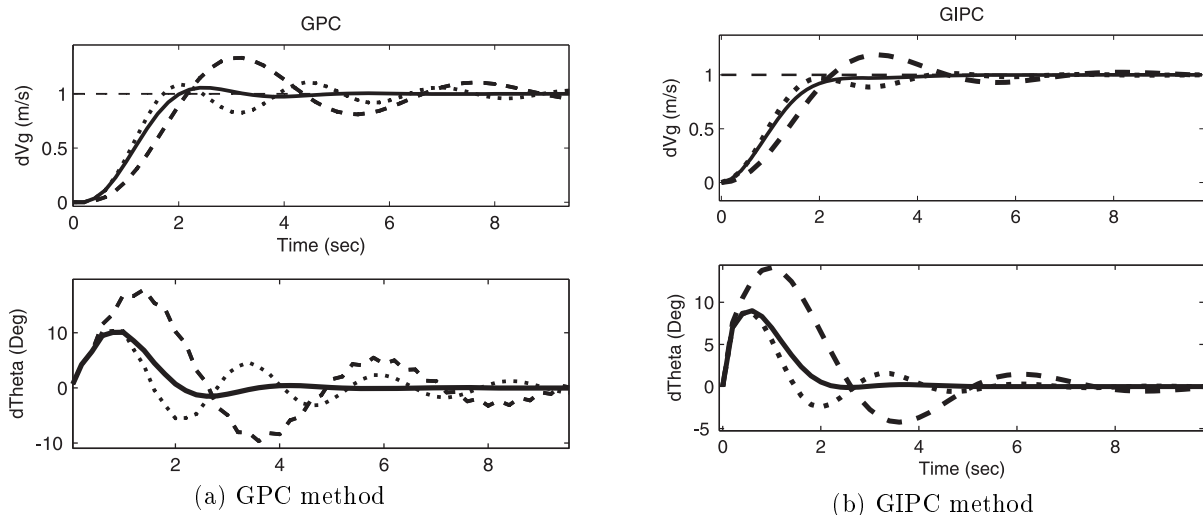


Figure 4. The step response and command of Q-guidance using GPC and GIPC for case A (solid line), case B (dashed line) and case C (dotted line).

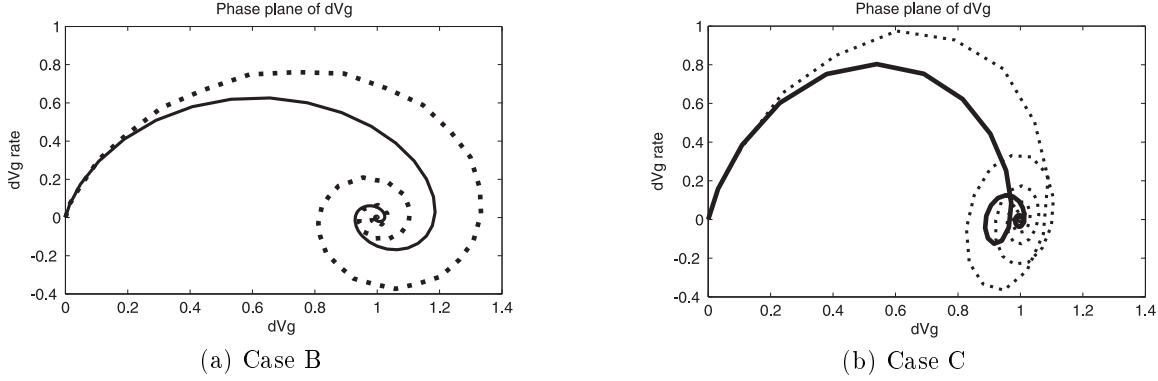


Figure 5. The phase plane of the deviated velocity to be gained using GPC (dotted line) and GIPC (solid line) methods for Case B and Case C. The initial and final values of the deviated velocity are 0.0 m/s and 1.0 m/s, respectively.

the required velocity minus the vehicle velocity, is the solution of the following differential equation:

$$\dot{\mathbf{V}}_g = -\mathbf{Q}\mathbf{V}_g - \mathbf{a} \quad (18)$$

where \mathbf{Q} is the matrix of partial derivatives of required velocity vector with respect to position vector, and \mathbf{a} is non-gravitational acceleration. Dotting \mathbf{V}_g into (18) gives:

$$\mathbf{V}_g^T \dot{\mathbf{V}}_g = -\mathbf{V}_g^T \mathbf{Q} \mathbf{V}_g - \mathbf{V}_g^T \mathbf{a} \quad (19)$$

but since:

$$V_g^2 = \mathbf{V}_g^T \mathbf{V}_g \quad (20)$$

Now, we define unit vector \mathbf{n}_g as follows:

$$\mathbf{n}_g = \frac{\mathbf{V}_g}{V_g} \quad (21)$$

Eq. (19) can be rewritten as:

$$V_g \dot{V}_g = -V_g^2 \left[\mathbf{n}_g^T \mathbf{Q} \mathbf{n}_g \right] - V_g a \cos \theta_g \quad (22)$$

or

$$\dot{V}_g = -\mathbf{n}_g^T \mathbf{Q} \mathbf{n}_g V_g - a \cos \theta_g \quad (23)$$

where a is the magnitude of \mathbf{a} , and θ_g is the angle between \mathbf{a} and \mathbf{V}_g vectors. The linearized state space form of Eq. (23) (guidance model) is:

$$\dot{x}(t) = A_c(t)x(t) + B_c(t)u(t) \quad (24)$$

where $x(t) \triangleq \delta V_g$, $u(t) \triangleq \delta \theta_g$, $B_c(t) \triangleq -a$ and $A_c(t) = -\mathbf{n}_g^T \mathbf{Q} \mathbf{n}_g$. Finally, the discrete state space form of Eq. (24) is written as:

$$x(k+1) = A(k)x(k) + B(k)u(k) \quad (25)$$

in which $A(k)$ and $B(k)$ are the time varying coefficients.

REFERENCES

1. Townsend G.E., Abbott A.S. and Palmer R.R., "Flight Mechanics and Trajectory Optimization", *Boost Guidance Equations*, NASA CR-1007, **8**, (1968).
2. Bhat M.S., Shrivastava S.K., "An Optimal Q-Guidance Scheme for Satellite Launch Vehicles", *Journal of Guidance Control and Dynamics*, **10**(1), PP 53-60(1987).
3. Battin R.H., *An Introduction to the Mathematics and Methods of Astrodynamics*, AIAA Inc., (1999).
4. Siouris G.M., *Missile Guidance and Control Systems*, Springer-Verlag Inc., (2004).
5. Shneydor N.A., *Missile Guidance Pursuit Kinematics, Dynamics and Control*, Horwood Publishing Limited, (1998).
6. Ben-Asher J.Z. and Belapolsky O., "On two formulations of linear quadratic optimal guidance", *AIAA Guidance, Navigation and Control Conference and Exhibit*, AIAA 2007-6533, South Carolina, (2007).
7. Bollino K.P., "Optimal guidance command generation and tracking for reusable launch vehicle reentry", *AIAA Guidance, Navigation and Control Conference and Exhibit*, AIAA 2006-6691, Colorado, (2006).
8. Cutler R., Ramaker B.L., "Dynamic Matrix Control: a Computer Control Algorithm", *In Proceedings of Joint Automatic Control Conference*, (1980).
9. Rouhani R. and Mehra R.K., "Model Algorithmic Control (MAC); Basic Theoretical Properties", *Automatica*, **18**(4), PP 401-414(1982).
10. Garcia C.E., Prett D.M. and Morari M., "Model Predictive Control Theory and Practice a Survey", *Automatica*, **25**(3), PP 335-348(1989).
11. Mayne D.Q., et. al., "Constrained Model Predictive Control: Stability and Optimality", *Automatica*, **36**, PP 789-814(2000).
12. Clarke D.W., Mohtadi C. and Tuffs P.S., "Generalized Predictive Control-Part I. The Basic Algorithm", *Automatica*, **23**(2), PP 137-148(1987).

13. Clarke D.W., Mohtadi C. and Tuffs P.S., "Generalized Predictive Control-Part II. Extensions and Interpretations", *Automatica*, **23**(2), PP 149-160(1987).
14. Mohtadi C., "On the Role of Prefiltering in Parameter Estimation and Control", *Workshop on Adaptive Control Strategies for Industrial Use*, Banff, Canada, (1988).
15. Clarke D.W. and Mohtadi C., "Properties of Generalized Predictive Control", *Automatica*, **25**(6), PP 869-876(1989).
16. Rossiter J.A., *Model Based Predictive Control: A practical Approach*, CRC press LLC, (2003).
17. Camacho E.F., *Model Predictive Control*, Springer-Verlag, London, (1999).
18. Maciejowski J.M., *Predictive Control with Constraints*, Prentice Hall, Harlow , (2002).
19. Balbis L., Katebi R., Dunia R., Ordys A., and Grimble M. J., "Nonlinear Predictive Control for Real Time Applications", *of the 2006 IEEE International Conference on Control Applications*, (2006).
20. Blet N., Megias D., Serrano J. and De Prada C., "Non-linear MPC Versus MPC Using On-Line Linearization-Comparative Study", *Proceedings of the 15th Terminal World Congress*, Barcelona, (2002).
21. Bitmead R.R., Gevers M., and Wertz V., *Adaptive Optimal Control, The Thinking Man's GPC*, Prentice Hall, (1990).
22. Chen R. H. and Speyer J.L., "Game-Theoretic Homing Missile Guidance with Autopilot Lag", *AIAA Guidance, Navigation and Control Conference and Exhibit, AIAA 2007-6535*, South Carolina, (2007).
23. Wagner J. T., "Cross Product Steering Implementation and Requirements on Thrust for Various Missions", *IEEE NAECON*, PP 122-132(1970).