Research Inventy: International Journal Of Engineering And Science
Vol.05, Issue 01 (January 2015), PP: 43-55
Issn (e): 2278-4721, Issn (p):2319-6483, www.researchinventy.com

Analytical Evaluation of Generalized Predictive Control Algorithms Using a Full Vehicle Multi-Body Dynamics Model For Mobility Enhancement

Ross Brown[1], Muthuvel Murugan*, Marcus Mazza[+]
U.S. Army Research Laboratory (U.S. ARL)-Vehicle Technology Directorate
Aberdeen Proving Ground, Maryland 21005, USA.
[1] Ex-Contract Research Engineer, U.S. ARL
** Corresponding Author, Research Aerospace Engineer, U.S. ARL*
[+] U.S Army Materiel Systems Analysis Activity (AMSAA), Aberdeen Proving Ground, MD 21005, USA.

ABSTRACT: *This paper discusses research conducted by the U.S. Army Research Laboratory (ARL) - Vehicle Technology Directorate (VTD) on advanced suspension control. ARL-VTD has conducted research on advanced suspension systems that will reduce the chassis vibration of ground vehicles while maintaining tire contact with the road surface. The purpose of this research is to reduce vibration-induced fatigue to the Warfighter as well as to improve the target aiming precision in-theater. The objective of this paper was to explore the performance effectiveness of various formulations of the Generalized Predictive Control (GPC) algorithm in a simulation environment. Each version of the control algorithm was applied to an identical model subjected to the same ground disturbance input and compared to a baseline passive suspension system. The control algorithms considered include a GPC with Implicit Disturbances, GPC with Explicit Disturbances, and GPC with Preview Control. A two-axle tactical vehicle with independent front and rear suspensions was modeled in the TruckSim full-vehicle dynamics simulator. The control algorithms were compared based on their effectiveness in controlling peak acceleration and overall average acceleration over a range of vehicle speeds. The algorithms demonstrated significant reductions in the chassis acceleration and pitch of the full-vehicle model.*

Keywords: *Vehicle vibration control, vehicle dynamics simulation, control algorithm, advanced suspension, generalized predictive control.*

I. INTRODUCTION

Passive suspension systems for Army wheeled vehicles are optimized for off-road passenger comfort and maneuverability, while maintaining acceptable on-road handling. These two goals often lead to conflicting design requirements. With a view toward available suspension systems that would improve both performance metrics, the Vehicle Technology Directorate (VTD) of the U.S. Army Research Laboratory (ARL) has underway a research program aimed at developing active suspension systems for implementation on the next generation of Army vehicles. The overall objective is to reduce vibration-induced fatigue as well as improve target aiming precision in-theater, while maintaining on-road handling. There are basically two main groups of active suspension systems: semi-active and fully active. Semi-active suspension systems vary spring and damper properties to store or remove energy from the system according to some applied control algorithm. Usually, this is achieved through variable damping components such as variable orifice valves or magneto-rheological fluids. Fully active suspension systems use force actuators to add and remove energy from the system. This can be achieved using hydraulic, pneumatic, or electromechanical actuators. Current research at VTD is addressing both types of systems. However, this paper focuses on fully-active controls and the application of a predictive control technique known as Generalized Predictive Control (GPC). GPC is a linear, time-invariant, multi-input/multi-output method that uses an Auto Regressive eXogenous (ARX) model to represent the input-output behavior of the system and to design the controller [3, 4, 8, 7, 9]. Three variants of the techniques have been evaluated. In the first variant, no explicit account is taken of the disturbances acting on the system during system identification and controller design [4, 8]. This version is called *GPC with Implicit Disturbance* in this paper. The second and third variants take explicit account of measurable disturbances. The first of these two (called *GPC with Explicit Disturbance* herein) uses only the past values of the measured disturbances while the latter (called *GPC with Preview* in this paper) additionally uses measurements (or predictions) of future disturbances [3, 7, 9]. The purpose of this paper is to present the results of numerical simulations of the application of the aforementioned GPC variants to a multi-body dynamic model of an Army tactical vehicle with nonlinear suspension systems. The paper begins with a summary of the key equations describing the underlying method including system identification, computation of the control law, and calculation of the commands sent to the actuators. Then, a brief discussion of the simulation environment and the vehicle model are presented. This is followed by a description of the three road profiles used as disturbances in the simulations. Representative results from the numerical simulations are then presented. The paper finishes with some concluding remarks on the results of the numerical study and comments on the nature of some planned future work.

II. CONTROL METHODOLOGY

The steps in developing a GPC-based controller are system identification and derivation of the control law. System identification is based on the derivation of a multi-step output prediction equation using the Observer Markov Parameters (OMP) that comprise the ARX model that is used to characterize the system. The control law then follows from the minimization of a quadratic performance equation involving the predicted response and the control command. A summary of the key steps of the process, adapted from the more extensive development present in reference [4], is given below. Since GPC is a multi-input/multi-output controller, the control command,$\{u_t\}$, measured response,$\{y_t\}$, and external disturbance, $\{d_t\}$ can be vectors whose elements correspond to the values of a specific actuator or sensor at time *t*.

GPC with Implicit Disturbance

The relationship between the input and output time histories of a linear, time-invariant multi-input/multi-output system can be described by an ARX model of the form:

$$\begin{aligned} y(t) = \alpha_1 y(t-1) + \alpha_2 y(t-2) + \ \dots\ + \alpha_p y(t-p) + \\ \beta_0 u(t)\ + \beta_1 u(t-1) + \ \dots\ + \beta_p u(t-p) \end{aligned} \tag{1}$$

This equation says that the output, *y(t)*, at the current time step, *t*, may be estimated by using *p* sets of the previous input and output measurements, *u(t-1), …, u(t-p), y(t-1),…,y(t-p)*, and the current input, *u(t)*. The integer *p* denotes the order of the ARX model. The coefficient matrices α_i and β_i are called Observer Markov Parameters and are the quantities computed by the identification process [2].

To start the identification process, the system is excited with band-limited white noise. These independent random excitations are applied concurrently to all of the r_c control inputs. The r_c control inputs and the *m* response outputs are recorded for *L* samples. The resulting input and output time histories, *u* and *y*, are then used to form the data matrices, *y* and *V*.

$$\boldsymbol{y} = \boldsymbol{\bar{Y}V} \tag{2}$$

where

$$\underset{m\,\mathrm{x}\,L}{\boldsymbol{y} = [y_0 \quad \cdots \quad y_{L-1}]} \tag{3}$$

and

$$\boldsymbol{V} = \underset{[r_c + (r_c+m)p]\,\mathrm{x}\,L}{\begin{bmatrix} u_0 & u_1 & u_2 & \cdots & u_p & \cdots & u_{L-1} \\ & u_0 & u_1 & \cdots & u_{p-1} & \cdots & u_{L-2} \\ & y_0 & y_1 & \cdots & y_{p-1} & \cdots & y_{L-2} \\ & & u_0 & \cdots & u_{p-2} & \cdots & u_{L-3} \\ & & y_0 & \cdots & y_{p-2} & \cdots & u_{L-3} \\ & & & \ddots & \vdots & \cdots & \vdots \\ & & & & u_0 & \cdots & u_{L-p-1} \\ & & & & y_0 & \cdots & y_{L-p-1} \end{bmatrix}} \tag{4}$$

Throughout this paper, the dimensions of the matrices and vectors are noted below key equations. The order of the ARX model, *p*, and the number of samples, *L*, are specified by the user. Some guidelines for their selection are given in reference [2]. The matrix $\bar{Y}$, containing the OMP, α_i and β_i , follows from equation 2 by inversion and has the form:

$$\underset{m\,\mathrm{x}\,[r_c + (r_c+m)p]}{\boldsymbol{\bar{Y}} = [\beta_0 \quad \beta_1 \quad \alpha_1 \quad \beta_2 \quad \alpha_2 \quad \dots \quad \beta_p \quad \alpha_p]} \tag{5}$$

The one-step-ahead output prediction equation given by equation 1 is the starting point for deriving the multi-step output prediction equation that is needed for deriving the GPC controller. This equation is obtained by replacing *t* by *t+j* in equation 1, and letting *j* range over the set of values *j=1, 2, … h_p-1*, where h_p is the prediction horizon (the number of time steps for which the future responses are predicted). The resulting equations can be assembled into a multi-step output prediction equation having the form:

$$\{y_{h_p}\} = \tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\} \tag{6}$$

The coefficient matrices τ, β, and α are formed from combinations of the OMP. The quantity y_{h_p} is the vector containing the future predicted responses, whereas u_{h_c} is the vector containing the (unknown) future control commands. The integer h_c is the control horizon (the number of time steps over which future control is assumed to act). The quantities u_p and y_p are vectors containing the previous *p* sets of control commands, and measured responses, respectively.

$$\underset{mh_p \text{ x } 1}{\begin{Bmatrix} y_{t+0} \\ \vdots \\ y_{t+h_p-1} \end{Bmatrix}} = \underset{mh_p \text{ x } r_c h_c}{\begin{bmatrix} \beta_0 & & & \\ \beta_0^1 & \beta_0 & & \\ \vdots & \vdots & \ddots & \\ \beta_0^{q-1} & \beta_0^{q-2} & \cdots & \beta_0 \\ \beta_0^{q} & \beta_0^{q-1} & \cdots & \\ \vdots & \vdots & \cdots & \\ \beta_0^{h_p-1} & \beta_0^{h_p-2} & \cdots & \beta_0^{h_p-h_c} \end{bmatrix}} \underset{r_c h_c \text{ x } 1}{\begin{Bmatrix} u_{t+0} \\ \vdots \\ u_{t+h_c-1} \end{Bmatrix}} +$$

$$\underset{mh_p \text{ x } mp}{\begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_p \\ \alpha_1^1 & \alpha_2^1 & \cdots & \alpha_p^1 \\ \vdots & \vdots & \cdots & \vdots \\ \alpha_1^{q-1} & \alpha_2^{q-1} & \cdots & \alpha_p^{q-1} \\ \alpha_1^{q} & \alpha_2^{q} & \cdots & \alpha_p^{q} \\ \vdots & \vdots & \cdots & \vdots \\ \alpha_1^{h_p-1} & \alpha_2^{h_p-1} & \cdots & \alpha_p^{h_p-1} \end{bmatrix}} \underset{mp \text{ x } 1}{\begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix}} + \qquad (7)$$

$$\underset{mh_p \text{ x } r_c p}{\begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_p \\ \beta_1^1 & \beta_2^1 & \cdots & \beta_p^1 \\ \vdots & \vdots & \cdots & \vdots \\ \beta_1^{q-1} & \beta_2^{q-1} & \cdots & \beta_p^{q-1} \\ \beta_1^{q} & \beta_2^{q} & \cdots & \beta_p^{q} \\ \vdots & \vdots & \cdots & \vdots \\ \beta_1^{h_p-1} & \beta_2^{h_p-1} & \cdots & \beta_p^{h_p-1} \end{bmatrix}} \underset{r_c p \text{ x } 1}{\begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix}}$$

where the coefficients are calculated using the following relations:

$$\beta_0^q = \beta_1^{q-1} + \alpha_1^{q-1}\beta_0 \qquad (8a)$$

$$\alpha_{p-1}^q = \alpha_p^{q-1} + \alpha_1^{q-1}\alpha_{p-1} \qquad (8b)$$

$$\beta_{p-1}^q = \beta_p^{q-1} + \alpha_1^{q-1}\beta_{p-1} \qquad (8c)$$

In this formulation the size of the ARX model, *p*, is set equal to the number of future predicted responses, h_p, (*prediction horizon*) and future commands, h_c, (*control horizon*). Generally, these three parameters can differ from one another [3, 8].

The optimal control law is obtained by minimizing an objective function. To accomplish this, one defines a quadratic objective function as follows,

$$\boldsymbol{J = y_{h_p}^T Q y_{h_p} + u_{h_c}^T R u_{h_c}} \qquad (9)$$

where

Q is a symmetric positive semi-definite matrix that assigns weights to the predicted responses,
R is a symmetric positive definite matrix that assigns weights on the future control commands,
The superscript *T* indicates the vector or matrix transpose.
Substituting the expanded equation for [*y*] yields,

$$\boldsymbol{J = [\tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\}]^T Q[\tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\}] + u_{h_c}^T R u_{h_c}} \qquad (10)$$

The minimum is obtained by taking the partial derivative of the function with respect to future commands, $\{u_{h_c}\}$ and setting it equal to zero.

$$\boldsymbol{0 = \frac{\partial}{\partial u_{h_c}}[\tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\}]^T Q[\tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\}] + \frac{\partial}{\partial u_{h_c}} u_{h_c}^T R u_{h_c}} \qquad (11a)$$

$$\boldsymbol{0 = 2\tau^T Q[\tau\{u_{h_c}\} + \alpha\{y_p\} + \beta\{u_p\}] + 2R\begin{Bmatrix} u_{t+0} \\ \vdots \\ u_{t+h_c} \end{Bmatrix}} \qquad (11b)$$

Solving for control commands, yields the following optimal future commands:

$$\boldsymbol{\{u_{h_c}\} = -(\tau^T Q\tau + \mathrm{R})^{-1}\tau^T Q[\alpha\{y_p\} + \beta\{u_p\}]}. \qquad (12)$$

Defining the following control law matrices,

$$\mathbf{A}_C = -(\boldsymbol{\tau}^T Q \boldsymbol{\tau} + \mathbf{R})^{-1} \boldsymbol{\tau}^T Q \boldsymbol{\alpha} \quad (13a)$$

$$\mathbf{B}_C = -(\boldsymbol{\tau}^T Q \boldsymbol{\tau} + \mathbf{R})^{-1} \boldsymbol{\tau}^T Q \boldsymbol{\beta} \quad (13b)$$

yields control laws that calculate the next h_c sets of commands,

$$\{u_{h_c}\} = \mathbf{A}_C\{y_p\} + \mathbf{B}_C\{u_p\} \quad (14)$$

where u_{h_c} corresponds to the control commands h_c time steps into the future.

However, only the first set of control commands will be applied to the system, thus only the first r_c rows of $\{u_{h_c}\}$ are needed. The remaining rows are discarded.

$$\underbrace{\{u_{t+0}\}}_{r_c \text{ x } 1} = \underbrace{\boldsymbol{\alpha}_C}_{r_c \text{ x } mp} \underbrace{\begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix}}_{mp \text{ x } 1} + \underbrace{\boldsymbol{\beta}_C}_{r_c \text{ x } r_c p} \underbrace{\begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix}}_{r_c p \text{ x } 1} \quad (15)$$

GPC with Explicit Disturbance

Generalized Predictive Controller with Explicit Disturbance, as discussed in [3, 9], is similar to *GPC with Implicit Disturbance*, except that now the external disturbances are explicitly measured and included in the model. Time histories of controls commands, measurement responses, and, now, disturbance measurements are used to generate the ARX model for the system. The V matrix with disturbance information is shown below.

$$V = \underbrace{\begin{bmatrix} d_0 & d_1 & \cdots & d_p & \cdots & d_{L-1} \\ u_0 & u_1 & \cdots & u_p & \cdots & u_{L-1} \\ & d_0 & \cdots & d_{p-1} & \cdots & d_{L-2} \\ & u_0 & \cdots & u_{p-1} & \cdots & u_{L-2} \\ & y_0 & \cdots & y_{p-1} & \cdots & y_{L-2} \\ & & \ddots & \vdots & \cdots & \vdots \\ & & & u_0 & \cdots & u_{L-p-1} \\ & & & y_0 & \cdots & y_{L-p-1} \end{bmatrix}}_{[r_d + r_c + (r_d + r_c + m)p] \text{ x } L} \quad (16)$$

The new matrix $\bar{Y}$, containing the OMP has the form:

$$\bar{Y} = \underbrace{[\delta_0 \quad \beta_0 \quad \delta_1 \quad \beta_1 \quad \alpha_1 \quad \delta_2 \quad \beta_2 \quad \alpha_2 \quad \dots \quad \delta_p \quad \beta_p \quad \alpha_p]}_{m \text{ x } [r_d + r_c + (r_d + r_c + m)p]} \quad (17)$$

This OMP values are used to predict future responses based on past responses (including external disturbances) and control commands.

$$\{y_{h_p}\} = \boldsymbol{\tau}\{u_{h_c}\} + \boldsymbol{\alpha}\{y_p\} + \boldsymbol{\beta}\{u_p\} + \boldsymbol{\delta}\{d_p\} \quad (18a)$$

$$\begin{Bmatrix} y_{t+0} \\ \vdots \\ y_{t+h_p} \end{Bmatrix} = \boldsymbol{\tau} \begin{Bmatrix} u_{t+0} \\ \vdots \\ u_{t+h_c} \end{Bmatrix} + \boldsymbol{\alpha} \begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix} + \boldsymbol{\beta} \begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix} + \boldsymbol{\delta} \begin{Bmatrix} d_{t-1} \\ \vdots \\ d_{t-p} \end{Bmatrix} \quad (18b)$$

As before, the next control commands are calculated by minimizing the objective function and taking the partial derivative with respect to control commands; the rows corresponding to future control commands are dropped. Using this extension, the measured disturbance information can be used.

$$\underbrace{\{u_{t+0}\}}_{r_c \text{ x } 1} = \underbrace{\boldsymbol{\alpha}_C}_{r_c \text{ x } mp} \underbrace{\begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix}}_{mp \text{ x } 1} + \underbrace{\boldsymbol{\beta}_C}_{r_c \text{ x } r_c p} \underbrace{\begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix}}_{r_c p \text{ x } 1} + \underbrace{\boldsymbol{\delta}_C}_{r_c \text{ x } r_d p} \underbrace{\begin{Bmatrix} d_{t-1} \\ \vdots \\ d_{t-p} \end{Bmatrix}}_{r_d p \text{ x } 1} \quad (19)$$

GPC with Preview Control

Preview Control or *Look-Ahead Control* measures the external disturbance before it affects the vehicle and incorporates this information into the control algorithms. Preview Control was first published in [1], and is further developed in [5, 6] for linear full-state feedback control.

Generalized Predictive Controller with Preview Control is a direct extension of *GPC with Explicit Disturbance*. If it is possible to generate a model of the explicit disturbance in order to predict future disturbances [9]; this future information can be used in conjunction with the control command, external disturbance, and measured response time histories to generate the future control commands. This research, however, assumes that the future disturbance can be directly measured using vehicle based sensors. Again, time histories of controls commands, measurement responses, and disturbance measurements are used to generate the ARX model for the system. The matrix $\bar{Y}$ is identical to equation (17). These are used to predict future responses based on past responses, past and future disturbances, and past and future control commands. In this formulation, it is assumed that the current and future disturbance can be measured. The matrix $\bar{Y}$ is the component of the ARX model that corresponds to the current and future disturbances.

$$\begin{Bmatrix} y_{t+0} \\ \vdots \\ y_{t+h_p} \end{Bmatrix} = \tau \begin{Bmatrix} u_{t+0} \\ \vdots \\ u_{t+h_c} \end{Bmatrix} + \gamma \begin{Bmatrix} d_{t+0} \\ \vdots \\ d_{t+h_c} \end{Bmatrix} + \alpha \begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix} + \beta \begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix} + \delta \begin{Bmatrix} d_{t-1} \\ \vdots \\ d_{t-p} \end{Bmatrix} \quad (20)$$

As before, the future control commands are calculated by minimizing the objective function and taking the partial derivative with respect to control commands; only the rows corresponding to the next immediate control commands are retained. Using this extension, both the future and the previous measured disturbance information can be used.

$$\{u_{t+0}\} = \gamma_C \begin{Bmatrix} d_{t+0} \\ \vdots \\ d_{t+h_c} \end{Bmatrix} + \alpha_C \begin{Bmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \end{Bmatrix} + \beta_C \begin{Bmatrix} u_{t-1} \\ \vdots \\ u_{t-p} \end{Bmatrix} + \delta_C \begin{Bmatrix} d_{t-1} \\ \vdots \\ d_{t-p} \end{Bmatrix} \quad (21)$$

The difference in control command between *GPC with Explicit Disturbance* and *GPC with Preview Control* is solely due to the presence of the future disturbance term. The control matrices, α_C, β_C, and δ_C are the same for both control laws.

Modeling and Simulation

TruckSim is a widely utilized, commercially available software tool used to simulate and analyze the dynamic behavior of wheeled vehicles. It is employed in conjunction with Matlab/Simulink in order to simulate the effects of the GPC algorithms on a full-vehicle model.

Figure 1 - TruckSim Modified HMMWV model on Perryman 3.

The TruckSim model takes the full vehicle and terrain into consideration, including: driver controls and steering, road geometry, suspensions, tires, and powertrain. Component parameters are input into TruckSim via constants, linear coefficients, non-linear tables and algebraic formulas.

The driver model controls the vehicle steering and throttle in order to follow a specified path and speed profile. TruckSim removes artificial vehicle constraints; therefore, vehicle speed and direction can deviate from the specified profile.

Full three-dimensional (3-D) road geometry is specified at 0.1 meter increments. The steering system captures the steer of each wheel due to the steering system geometry and compliances. The suspension system models capture the full non-linear kinematical behaviors and compliances of the front and rear suspensions. These models include data for springs, dampers, and jounce/rebound stops. The tire model is an internal table-based, single point contact model. Non-linear tables using actual measured tire data represent vertical force, lateral force, longitudinal force, aligning moment, and overturning moment as functions of deflection, slip, load, and camber. The powertrain model includes engine torque, torque convertor characteristics, and transmission gear ratios and efficiencies. A representative TruckSim tactical vehicle model is shown in Figure 1.

TruckSim provides a generalized two-axle tactical vehicle model (HMMWV) with an independent front and rear suspension system. Figure 2 contains a graphical layout of the suspension as modeled in SuspensionSim.

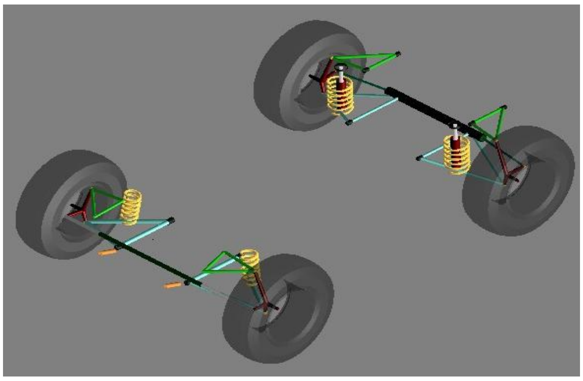


Figure 2 - Representative HMMWV Suspension

For this research, we modified some of the basic HMMWV vehicle parameters to study the performance of GPC algorithms on various terrain profiles. The vehicle mass was doubled. The suspension springs were stiffened, and the front damping coefficient was reduced. This is shown in Table 1.

Table 1 - Vehicle parameters for modified HMMWV model

	Basic HMMWV	Modified HMMWV
Sprung Mass	2210 kg	5000 kg
Front spring	150 N/mm	250 N/mm
Front damper	30 kNs/m	15 kNs/m
Rear spring	150 N/mm	400 N/mm
Rear damper	30 kNs/m	30 kNs/m

The equations of motion in the TruckSim math models are ordinary differential equations (ODE) derived from the first principles for full nonlinear 3-D motions of multiple connected rigid bodies. The sprung mass of the vehicle is a rigid body with six degrees of freedom (DOF). Each wheel has one spin DOF. Each independent suspension axle has two multi-body DOF for the vertical movements of the wheel centers. Other suspension motions such as camber, lateral position, etc., are constrained as nonlinear functions of the independent variables. Each suspension has six compliance DOF. Each tire has two dynamic DOF, one for lagged lateral slip and one for lagged longitudinal slip.

Because the GPC algorithms are implemented in Matlab and Simulink, TruckSim uses co-simulation to update the math model and the algorithm calculations simultaneously. TruckSim provides an S-function block to communicate with Simulink. The control algorithms run in Simulink using the discrete integration method with a fixed time-step of 0.001 seconds (1000 Hertz). TruckSim uses the Adams-Moulton second-order integration (AM-2) method; the time-step for this integration method is fixed at half the Simulink simulation rate. The model is integrated before and after each communication step with Simulink.

Within Simulink, the GPC algorithms sample the vehicle sensors once every 0.01 seconds (100 Hz). The actuator commands to TruckSim are, also, updated every 0.01 seconds. For this work, the control algorithms optimize commands for force actuators at each of the four suspension corners to minimize the sprung mass motion. TruckSim uses the External Springs option to introduce outside forces from the actuators. Additionally, the absolute road heights at the tire contact patch and at the preview sensor location are sent to the algorithms.

Road Profiles :

Three road profiles were used to evaluate the algorithms: Sine, Curb, and Perryman 3. In each case, the road profiles were provided as an array of ground distances and vertical heights.

Sine

This road profile was defined by a sine wave with a half-amplitude of 0.10 meters and a wavelength of 10 meters. The intent was to provide a single-frequency disturbance that relates to vehicle speed, in this case, 1 meter/second (3.6 kph) of speed corresponds to 0.1 Hz excitation. There is a 30 meter flat stretch to allow the model to reach steady state after startup. Figure 3 shows a portion of this type of road profile.

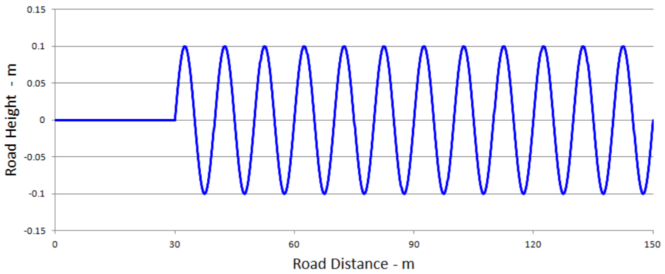


Figure 3 - Sine road profile.

Curb

This road profile is defined by a 0.10 meter step rise followed by 10 meters of level surface, and then a 0.10 meter step drop, as shown in Figure 4. The intent was to provide a step input disturbance into the system. There is a 30 meter flat stretch to let the initial oscillations die out and the vehicle reach steady state.

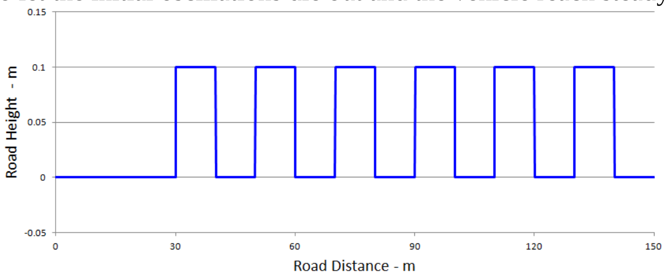


Figure 4 - Curb road profile.

Perryman3

Perryman 3 is an off-road test track located at the Aberdeen Proving Ground. This road profile, shown in Figure 5, is intended to evaluate the algorithms against a challenging, "real-world" course.

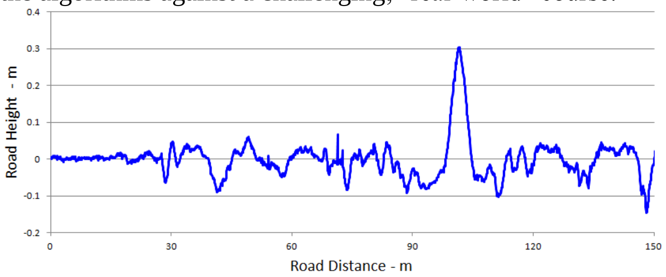


Figure 5 - Perryman 3 road profile.

Simulation Results

Simulations were run until the vehicle traveled 150 meters. The open loop model was run in parallel with closed-loop *GPC with Implicit Disturbance*, *GPC with Explicit Disturbance*, and *GPC with Preview Control* for performance comparison studies. All measurements assumed no noise or offset-bias. Initial conditions for the suspension system state variables were set to zero. Vehicle speed was held at a constant value for the entire run.

For this work the system identification for each algorithm was pre-computed. Band-limited white noise was used to excite the four actuators. The vehicle was run over Perryman 3 terrain profile at 40 kph. This provided broadband random disturbance inputs out to 50 Hertz with large peaks from 0.5 to 3 Hertz.

The following objective functions were used as metrics in several comparisons of algorithm performance:

$$J_{Resp} = low\,pass(\sqrt{y^T y}) \qquad (22a)$$

$$Actuator\ Force = low\,pass(\sqrt{u^T u}) \qquad (22b)$$

$$Actuator\ Power = low\,pass(\sqrt{(power^T power}) \qquad (22c)$$

where "lowpass" denotes a 4th order digital low-pass Butterworth filter with a cutoff frequency of 1 Hz.

Using the cosine double angle formula, it can be shown that the square of a signal will have a steady offset component and a high frequency component that is twice the frequency of the original signal. The low-pass filter removes the oscillatory component, leaving only the steady offset, which is related to the amplitude of the fast-fourier transform of the original signal. Because the cutoff frequency is higher than the lowest frequency of excitation in the road profiles, there is a small low frequency oscillation in the objective function, J. The low frequency oscillation is removed by averaging the objective function over the entire run-time.

For each simulation, the averaged values of J were plotted for a series of vehicle speeds up to 80 kph. The closed-loop responses were normalized by the corresponding values of the open-loop responses at each speed.

$$\boldsymbol{Response\ Improvement} = 1 - \frac{J_{Resp,algorithm}}{J_{Resp,open\ loop}} \tag{23}$$

This normalization allows a more meaningful comparison of the algorithms despite the variation in actuator force or power.

$$\begin{matrix}\boldsymbol{Force\ Normalized} \\ \boldsymbol{Effectiveness}\end{matrix} = \frac{Response\ Improvement_{algorithm}}{Actuator\ Force_{algorithm}} \tag{24a}$$

$$\begin{matrix}\boldsymbol{Power\ Normalized} \\ \boldsymbol{Effectiveness}\end{matrix} = \frac{Response\ Improvement_{algorithm}}{Actuator\ Power_{algorithm}} \tag{24a}$$

Simulation results based on the Modified HMMWV parameters are presented in Figures 6 through 12. These figures were generated using the following algorithm parameters, as shown in Table 2.

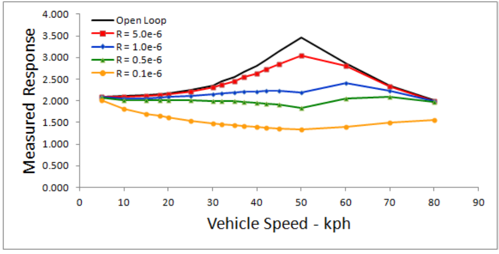
Table 2 - Algorithm tuning parameters used for modified HMMWV vehicle.

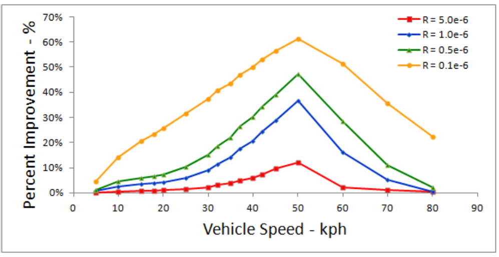
	GPC with Implicit Disturbance	GPC with Explicit Disturbance	GPC with Preview Control
actuators, r_c	4		
disturbances, r_d	4		
sensors, *m*	3		
Q	50 *(vertical acceleration)*, 10 *(pitch)*, 1 *(roll)*		
R	range { 1e-7 to 5e-6 }		
ARX model size, *p*	8		
Prediction Horizon, h_p	8		
Control Horizon, h_c	8		
Preview Horizon	-	-	8 (0.08s)
ID sample size, *L*	400 points		
Disturbance for ID	Perryman 3 at 40 kph		
Actuator ID signal	white noise		
Actuator Limit	1.5e5 N		

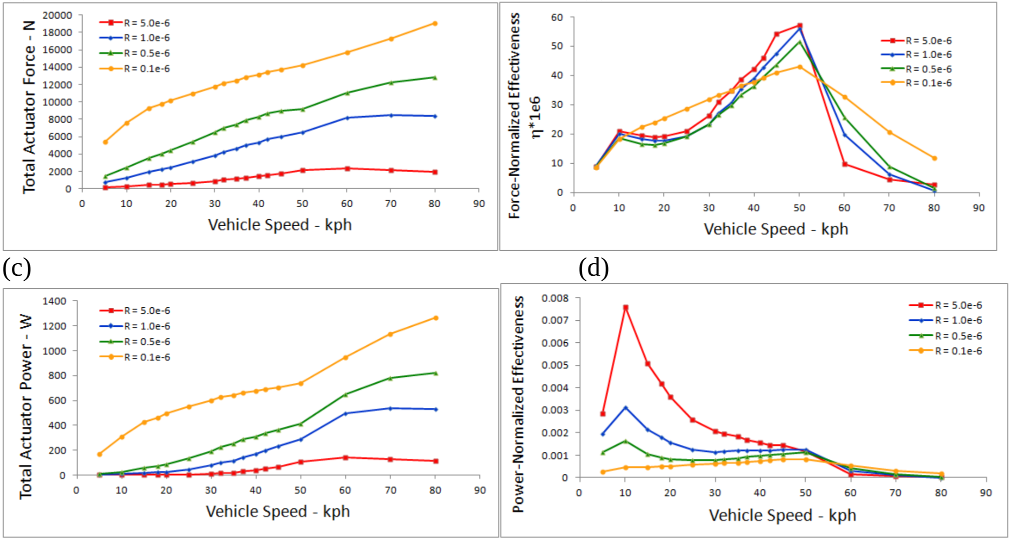
Effects of Control Weighting

The *modified HMMWV* model was run with *GPC with Preview Control* over the *Sine* road profile from 0 to 80 kph. The objective function, actuator force, and actuator power are averaged over a 150-meter simulation. The speed range corresponds to a sinusoidal excitation of 0 to 2.2 Hz. This is repeated several times for a range of control weights (R matrix values); the response weights (Q matrix values) were held constant. This effect of the Control Weighting matrix is shown in Figures 6(a) through 6(f).

An order of magnitude decrease in the control weight matrix produces a significant improvement in response and a corresponding increase in the required force and power. The force normalized effectiveness shows that the curves for each weight coalesce in the same regions; the exception of R = 0.1e-6 occurs because the forces required are at the limit for the actuator. This suggests that as long as the commanded forces are within the capability of the actuator, the level of the response is linear with control loads. However, when comparing the actuator power, the power-normalized effectiveness decreases as the control weights decreases. This shows that the actuator power costs increase much faster than the responses decrease. Actuator power, then, becomes the driving factor for choosing control weights.







(a) (b)

(c) (d)

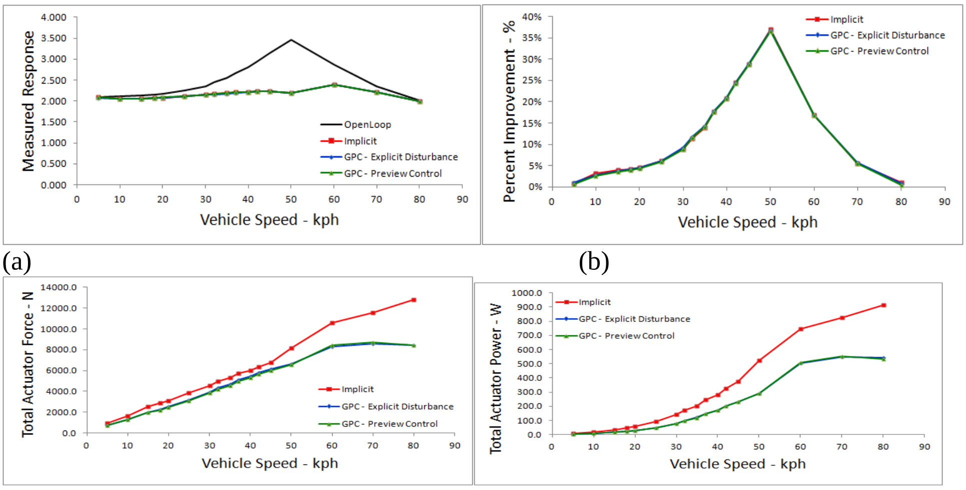
(e) (f)

Figure 6 - Effect of control weights on GPC with Preview Control on sine road profile.

Comparison of Algorithms

A short comparison of the three control laws operating on the three road courses (disturbance inputs) is presented in the discussion below. The control weight for *GPC with Explicit Disturbance* was R=1.0e-7 over the entire speed range. To simplify analysis, the control weights for the other two algorithms were chosen such that the magnitude of the measured responses matched *GPC with Explicit Disturbance* at each speed. Again, the objective function and the actuator force are averaged over the 150-meter run for sinusoidal disturbances from 0 to 80 kph (corresponding to 0 to 2.2 Hz). Figures 7(a) through 7(d) show a comparison of the three algorithms versus vehicle speed.

The open loop response reaches a peak at 50 kph; representing the vehicle's pitch mode resonance frequency. The closed-loop algorithms reduce the peak by 60%; effectively attenuating the natural frequency. The normalized data shows that, *GPC with Explicit Disturbance* and *GPC with Preview Control* have similar performance; requiring the same actuator force and power. However, *GPC with Implicit* requires significantly greater actuator force and power, almost twice the actuator power at 80 kph. The addition of road disturbance information to the GPC algorithm greatly reduces the actuator loads for the closed loop system.



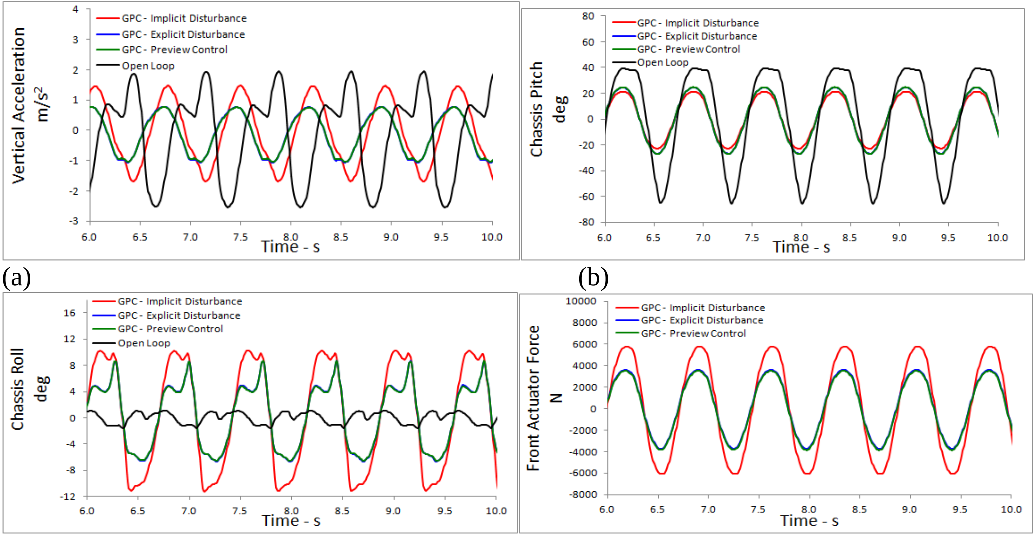
(a) (b)

(c) (d)

Figure 7 - Effect of GPC algorithms on sine road profile.

Figures 8(a) through 8(d) show the time-history of the measured responses and actuator forces for each algorithm as the vehicle maneuvers over the *Sine* road profile at 50 kph, which represents a single frequency disturbance of 1.4 Hz.

All three versions of the GPC algorithm reduce the chassis vertical acceleration and pitch. Chassis roll increases, due to the very low response weight assigned to chassis roll. *GPC with Implicit Disturbance* performs worse on controlling chassis vertical acceleration, but slightly better at reducing chassis pitch. Since the chassis pitch is the dominant term in the objective function, this results in similar levels for the response objective function. It is clear that *GPC with Implicit Disturbance* requires much higher actuator loads than the other two algorithms.

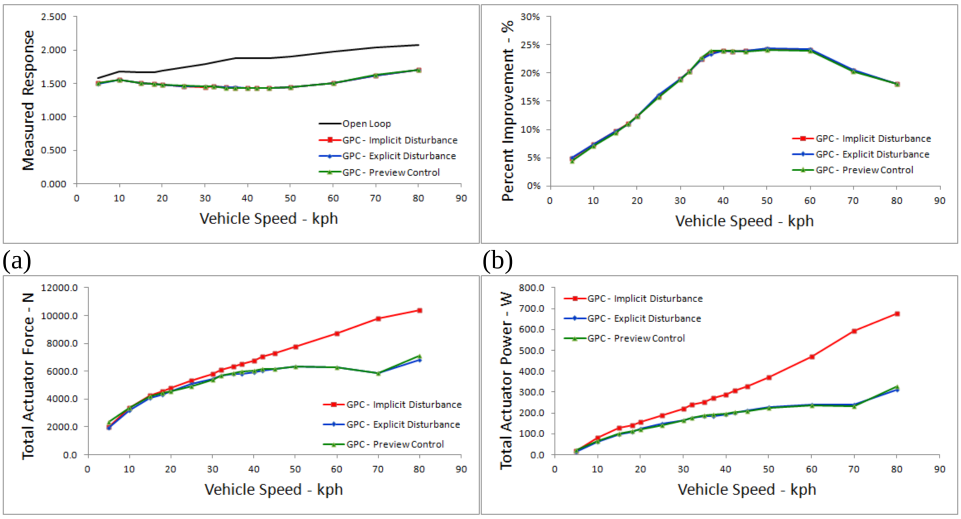


(a) (b)

(c) (d)

Figure 8 - Time History of Control Algorithms for Light Vehicle parameters on Sine course at 50 kph.

The same simulations were run using the Perryman 3 road profile; the algorithm performance is shown in Figures 9(a) through 9(d). The GPC algorithms reduce the overall response by more than 25% from the highest open loop response. Much like the performance on the Sine road profile, *GPC with Explicit Disturbance* and *GPC with Preview Control* have almost identical response and actuator objective functions. Once again, *GPC with Implicit Disturbance* requires significantly higher actuator forces and loads to match the response objective function.

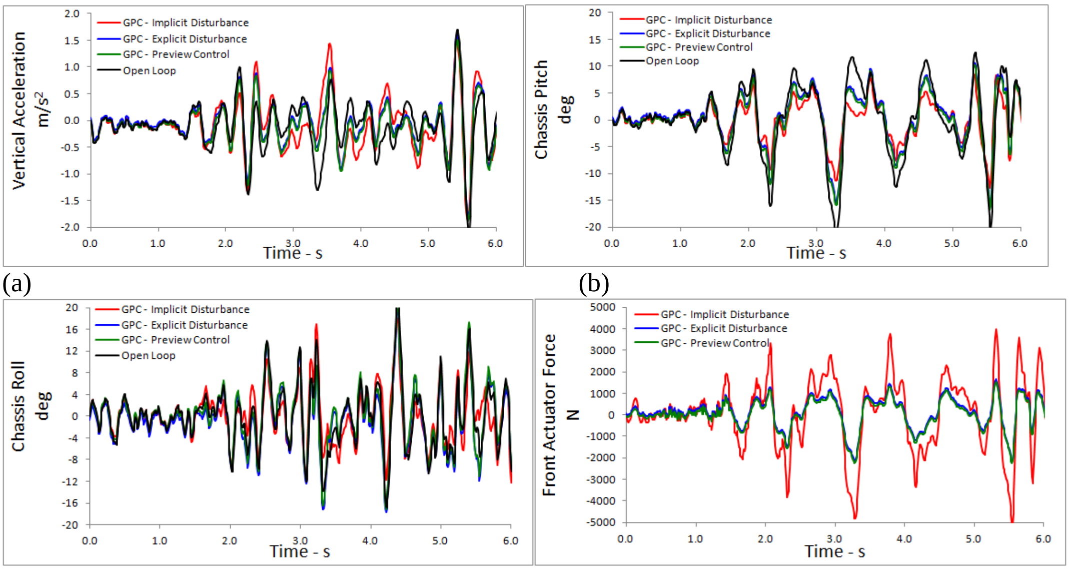


(a) (b)

(c) (d)

Figure 9 - Effect of GPC algorithms for Modified HMMWV on Perryman 3 road profile

The GPC algorithms reduce the peaks of the chassis pitch compared to open loop cases. This dominant sensor makes up for the slight increases in closed-loop roll and vertical acceleration. Without measured knowledge of the disturbance, *GPC with Implicit Disturbance* must wait until the sign of the responses change before applying a countering force. This results in the large, but brief forces as the algorithm catches up to the changing slope of the disturbance. The algorithms with disturbance information have a much quicker response, thus reducing the peaks. The time history of the vehicle on Perryman 3 is shown in Figures 10(a) through 10(d).

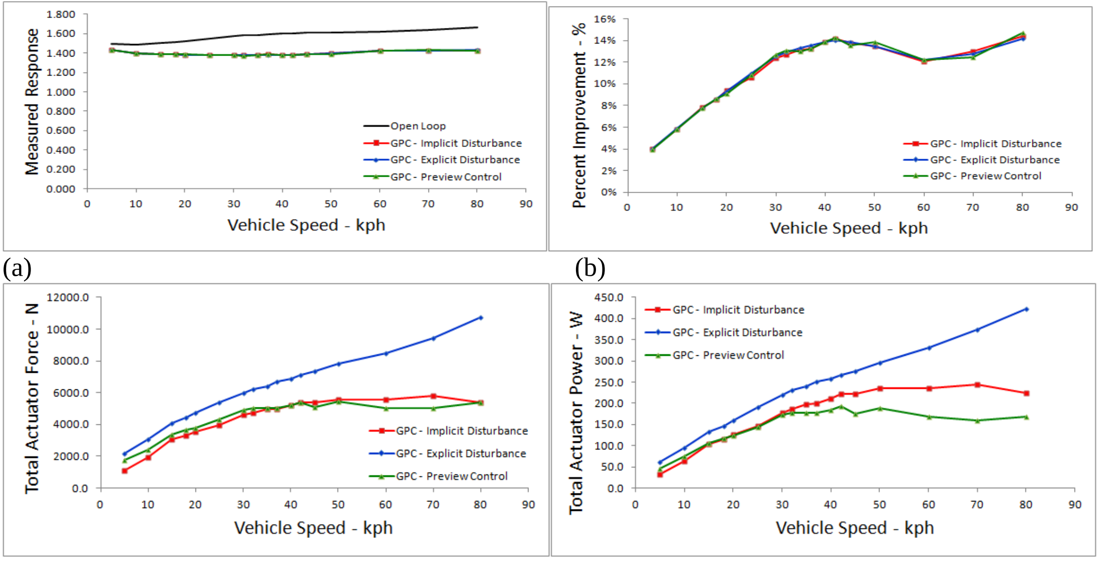


(a) (b)

(c) (d)

Figure 10 - Time History of GPC Algorithms for Modified HMMWV parameters on Perryman 3 course at 50 kph

The performance of the algorithms for the *Curb* road profile is shown in Figures 11(a) through 11(d). Again, all three GPC algorithms reduce the response objective function, by as much as 14% at the highest speed. However, this time, there is a surprising difference between the actuator force and power of the algorithms. *GPC with Explicit Disturbance* requires the highest actuator loads, double the force and power as required for *GPC with Implicit Disturbance*. *GPC with Preview Control* requires the least power of the three variants; less than 200 Watts above 50 kph as shown in Figure 11(d).



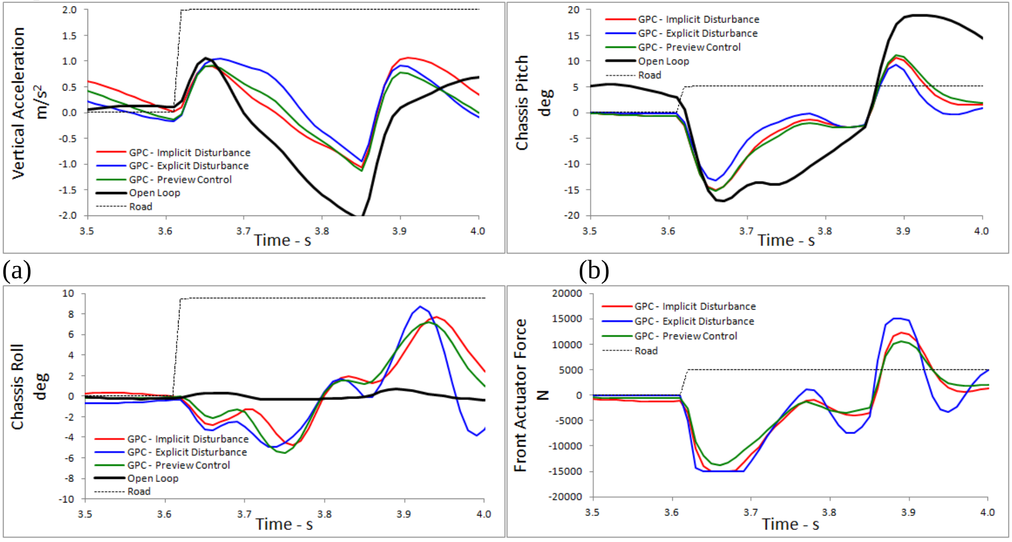
(a) (b)

(c) (d)

Figure 11 - Effect of GPC algorithms on curb road profile.

Previously, *GPC with Explicit Disturbance* was nearly indistinguishable from *GPC with Preview Control*. Figures 12(a) through 12(d) provide a closer look at the actuator forces experienced as the vehicle encounters the step height increase of the road. When the *GPC with Explicit Disturbance* encounters a step increase in the disturbance height (basically, a very steep slope), this steep slope produces a high control command. The steep slope is in the disturbance history for 8-timesteps. On the other hand, *GPC with Preview Control* issues small commands (100 N) for the 8-timesteps prior to the start of the curb. This reduces the peak force required once the tire contacts the curb, resulting in the lowest peak force. Additionally, after the tire contacts the curb, the future disturbance contains 8 time-steps of flat data to further temper the control commands.

As before, *GPC with Implicit Disturbance* is unaware of the disturbance and can only respond after the measured responses are affected by the curb. Thus, it commands high peaks, but not in response to an infinite slope.



(a) (b)

(c) (d)

Figure 12 - Time History of GPC Algorithms for Light Vehicle parameters on Curb course at 72 kph.

Concluding Remarks

The results of numerical simulations directed at evaluating the effectiveness of a predictive control method known as Generalized Predictive Control (GPC) for active control of chassis suspension systems in Army wheeled vehicles have been summarized. GPC is a linear time-invariant, multi-input/multi-output predictive control method that uses an ARX model to describe the input-output relationship characterizing the system. The coefficient matrices of the ARX equation are determined using system identification techniques and then used to form the matrices comprising the GPC control law. Key equations in the formulation of the method were summarized. Three variants of the basic GPC algorithm were investigated. The first assumes that the disturbances are unknown and computes the control law taking implicit account of the disturbances acting on the system (*GPC with Implicit Disturbance*). The second assumes that the past values of the disturbances can be measured and uses that information in deriving the control law (*GPC with External Disturbance*). The third variant assumes that the disturbances ahead of the vehicle can be measured and additionally uses that data when deriving the control law (*GPC with Preview Control*). Closed-loop behavior of the system was simulated using a full-vehicle nonlinear model of a two-axle tactical vehicle moving over three types of road profile disturbances for each of the three variants of GPC and compared to the open-loop behavior of the system. These comparisons show significant reductions in chassis motions of the model. Through simulation, it has been shown that, for a given algorithm, the control weights reduce sensor objective function at the same rate that it increases actuator objective function. Thus the control weight can be directly used to scale the total force available to a controller. *GPC with Implicit Disturbance* consistently requires higher actuator forces and more power than *GPC with Preview Control* or *GPC with Explicit Disturbance* when given a tonal or broad band disturbance. *GPC with Preview Control* performs nearly identically to baseline *GPC with Explicit Disturbance*; however, the preview control algorithm does drastically reduce the initial pulse due to step inputs. Despite the presence of disturbance information, *GPC with Explicit Disturbance* performs poorly on step road profile, even worse than having no disturbance information at all. In summary, for the three road course profiles investigated, *GPC with Preview Control* performs better overall than the other GPC variants for reducing vehicle-chassis acceleration and pitch. The simulation results show that *GPC with Preview Control* is capable of reducing the vehicle-chassis acceleration by as much as 25% on average for the most challenging terrain profile course simulated. The differences in the performance of the various algorithms studied in this research effort are currently being examined further to understand the main causes for the observed performance differences. In future, these GPC algorithms will be tested using a two-degree-of freedom suspension test rig at the Army Vehicle Research Laboratory to verify and validate these algorithms.

REFERENCES

[1] E.K. BENDER, OPTIMUM LINEAR PREVIEW CONTROL WITH APPLICATION TO VEHICLE SUSPENSION, AMSE JOURNAL OF BASIC ENGINEERING VOL 90, 1968.

[2] J-N. JUANG, APPLIED SYSTEM IDENTIFICATION. PTR PRENTICE HALL (1994).

[3] J-N. JUANG AND K.W. EURE, PREDICTIVE FEEDBACK AND FEEDFORWARD CONTROL FOR SYSTEMS WITH UNKNOWN DISTURBANCES. NASA TECHNICAL MEMORANDUM, 1998-208744, 1998.

[4] J-N. JUANG AND M.Q. PHAN, IDENTIFICATION AND CONTROL OF MECHANICAL SYSTEMS, CAMBRIDGE UNIVERSITY PRESS (2001).

[5] M.A.H. VAN DER AA, CONTROL CONCEPT FOR A SEMI-ACTIVE SUSPENSION WITH PREVIEW USING A CONTINUOUSLY VARIABLE DAMPER. PHD THESIS (1994), EINDHOVEN UNIVERSITY OF TECHNOLOGY.

[6] M.M. EL MADANY, Z. ABDULJABBAR, AND M. FODA, OPTIMAL PREVIEW CONTROL OF ACTIVE SUSPENSIONS WITH INTEGRAL CONSTRAINT. JOURNAL OF VIBRATION AND CONTROL, VOL 9:1377-1400, 2003.

[7] R. BROWN, J. PUSEY, M. MURUGAN, D. LE, GENERALIZED PREDICTIVE CONTROL ALGORITHM OF A SIMPLIFIED GROUND VEHICLE SUSPENSION SYSTEM, SAGE JOURNAL OF VIBRATION AND CONTROL, PAPER NO: 1077546312448505, 2012.

[8] R.G. KVATERNIK, J-N. JUANG AND R.L. BENNETT, EXPLORATORY STUDIES IN GENERALIZED PREDICTIVE CONTROL FOR ACTIVE AEROELASTIC CONTROL OF TILTROTOR AIRCRAFT. NASA TECHNICAL MEMORANDUM, 2000-210552, 2000.

[9] R.G. KVATERNIK, K.W. EURE, AND J-N. JUANG, EXPLORATORY STUDIES IN GENERALIZED PREDICTIVE CONTROL FOR ACTIVE GUST LOAD ALLEVIATION. NASA TECHNICAL MEMORANDUM, 2006-214296, 2006.

Nomenclature

ARL	U.S. Army Research Laboratory
ARX	AutoRegressive eXogenous input
d	disturbance inputs
F	control actuator force
GPC	Generalized Predictive Control
$\mathbf{h_p, h_c}$	prediction and control horizons
Hz	Hertz
kph	kilometers per hour
L	number of samples used for system identification
LVDT	Linear Velocity Displacement Transducer
m	meter
$\mathbf{m, r_c, r_d}$	number of response outputs, control inputs, disturbance inputs
N	Newton
OMP	Observer Markov Parameters
p	order of the ARX model
Q, R	response penalty, control penalty
s	seconds
t	index of current time step
u	control inputs
u(t), y(t), d(t)	vectors of current inputs, outputs, and disturbances at time step *t*
u(t-j) ,y(t-j), d(t-j)	vectors of past inputs, outputs, and disturbances at time step (*t-j*)
u(t+j) ,y(t+j), d(t+j)	vectors of future inputs, outputs, and disturbances at time step (*t+j*)
V	system identification data matrix
VTD	Vehicle Technology Directorate
y	response outputs
$\bar{Y}$	matrix of observer Markov parameters
$\alpha_i, \beta_i\ \delta_i$	observer Markov parameters
$\alpha_c, \beta_c, \gamma_c, \delta_c$	control law gain matrices
$\tau, \alpha, \beta, \gamma, \delta$	coefficient matrices in multi-step output prediction equation