September 2011

RDTA – RS

TECHNICAL PAPER

Tracked Vehicle Dynamics Modeling and Simulation Methodology, with Control, using RecurDyn Software Package

By:

Joseph Raymond

Vehicle Dynamics and Structures Team

CASSI – Analytics

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **01 SEP 2011**	2. REPORT TYPE **Technical Report**	3. DATES COVERED **01-09-2011 to 01-09-2011**

4. TITLE AND SUBTITLE **Tracked Vehicle Dynamics Modeling and Simulation Methodology, with Control, using RecurDyn Software Package**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Joseph Raymond**	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **U.S. Army TARDEC ,6501 E.11 Mile Rd,Warren,MI,48397-5000**	8. PERFORMING ORGANIZATION REPORT NUMBER **#22386**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **U.S. Army TARDEC, 6501 E.11 Mile Rd, Warren, MI, 48397-5000**	10. SPONSOR/MONITOR'S ACRONYM(S) **TARDEC** 11. SPONSOR/MONITOR'S REPORT NUMBER(S) **#22386**

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Many Army vehicles require tracks in order to meet the tough mobility requirements for the Army mission profile. Modeling and Simulation (M&S) provides a large cost-savings and offers a quick turn-around when addressing vehicle performance issues. Once a baseline model is built for a given system, the model can be changed quickly to address different load or usage profiles and to determine the overall affect on the vehicle and its performance. Tracked vehicles present a number of challenges, however, due to the large number of interactions between all the track and suspension components. Prior methods for analyzing tracked vehicle performance through M&S led to either simplified vehicle models or very long compute times due to the level of detail required to properly model tracked vehicles.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**12**	

1. Executive Summary

Many Army vehicles require tracks in order to meet the tough mobility requirements for the Army mission profile. Modeling and Simulation (M&S) provides a large cost-savings and offers a quick turn-around when addressing vehicle performance issues. Once a baseline model is built for a given system, the model can be changed quickly to address different load or usage profiles and to determine the overall affect on the vehicle and its performance. Tracked vehicles present a number of challenges, however, due to the large number of interactions between all the track and suspension components. Prior methods for analyzing tracked vehicle performance through M&S led to either simplified vehicle models or very long compute times due to the level of detail required to properly model tracked vehicles.

RecurDyn offers a number of potential benefits by using a recursive dynamic formulation which takes advantage of the fact that every track shoe is the same and that each track segment is connected to one another in the same manner. The software exploits this symmetry to greatly reduce model development times, complexity, and computational run times. RecurDyn features a user-friendly, graphical user interface (GUI), or front end, and a track-building toolkit (a.k.a. Trackbuilder). This front end saves time building the models by eliminating repeated processes, allowing the user to define one track segment and repeat it around the track loop. The software package also includes a built-in control program called CoLink, which can be used to control and drive vehicle models.

A methodology and library of standard templates were developed by the author to enhance the usability of RecurDyn specific to tracked vehicles. The end result is a dramatic decrease in tracked vehicle model build and run times, which in many cases makes simulation a faster, more cost effective option than the build-test-break-fix-test cycle of the past. These methodologies and templates are intended to serve as a reference for future TARDEC engineers learning to model tracked vehicles. To date, these templates include: path following terrain profiles, NATO double lane change, side slopes, performance on grades, and steady state turning circles. Additional events will be programmed as needed and added to the library. The final conclusion of this effort is that RecurDyn is both useful and powerful, and that RecurDyn may be the future for 3-Dimensional multi-body dynamics M&S work for tracked vehicles.

2. Contents

3. Problem Statement

Current TARDEC tracked vehicle Modeling and Simulation (M&S) tools either oversimplify the model or are cumbersome and time consuming to use. A current method of modeling involves combining all track and suspension components into a single representative element. Unfortunately, this method decreases the amount of data available to be measured, as an oversimplification of the system results in the removal of essential vehicle components from the model (track pads, track pins, shocks, track shoes, springs, etc.). Alternatively, all individual track components can be modeled independently and combined to approximate a single track segment. This single track segment is then copied, repositioned and combined to form the overall track. This effort is currently done with a text-based interface, and much scripting and computer programming knowledge is needed to build simple models. A hindrance to troubleshooting is inherent in the process – the user cannot start troubleshooting until the full model is complete and is simulated. This current process is time consuming, cumbersome, and is not easy for new users to learn.

The RecurDyn software package offers a validated solution (1) (2) (3). RecurDyn uses a specialized, hierarchical modeling approach with recursive algorithms for similar elements, such as with the track assembly. Basic models can be generated very quickly with fewer errors in the modeling process without loss of model fidelity. To develop a tracked vehicle model, the user enters several key track-segment parameters and the locations and geometries of the road-wheels, idler, sprocket, any idler-wheels and the basic track path around these items. The program then assembles the track around these items as a subassembly of the whole vehicle model. This subassembly can be simulated independently from the overall model which greatly reduces troubleshooting time because each subassembly can be isolated and ran during troubleshooting.

3.1 Strategic Background and Outlook

Implementation of a fast, easy dynamic simulation method for tracked vehicles has been a desire for TARDEC for decades. The Center for Simulation and Design Optimization of Mechanical Systems, based at the University of Iowa, was founded on grant money from the US Government, and led to the development of the Dynamic Analysis and Design System (DADS) multi-body dynamics simulation code. Throughout the 1980s, several shortcomings of the DADS package were noted and fixed. One particular issue with DADS, which still exists in the version of DADS implemented at TARDEC today, is with modeling track vehicles. Originally, each body within the track segment was modeled individually, with bushing/spring elements connecting each body within a given track segment, with pins connecting each track segment. The modeler must align each segment properly with the track pins with the sprocket teeth and build contacts between the track with the applicable road wheels, sprockets, idlers and idler wheels (4). As a result, this led to weeks of model-building time and days of (1980's) computer time per simulation run. (Note: in the mid 1980's, DADS was commercialized by the code developers who formed the company Computer Aided Design Software Inc. (CASDSI). In the late 1990's, CADSI was bought by LMS Inc., who integrated DADS into their "Virtual Lab" simulation suite. A GUI and a recursive element formulation was eventually added to DADS to build the track, but the GUI had issues which diminished its usability (4)).

Researchers at the University of Iowa worked to resolve this issue under a US Army contract (5). One solution was to build a single element called the Track Super Element (5). The Track Super Element was, in essence, a single, giant, flexible band. The implementation of the Track Super Element resulted in a loss of data fidelity and was not able to model important track load data (4), amongst other lost capabilities. Tracks, like tires, need to be replaced often. The track load data is vital to performing fatigue analyses on tracks to determine how often they should be replaced, which is a very important to the overall reliability and readiness of track vehicles.

Alternatively, researchers at the University of Iowa also developed a recursive dynamic formulation in which the modeler builds a representative track segment (6). The representative track segment is copied and linked together using a simplified algorithm which assumes each track segment is identical, with force/torque pairs at each linkage. An analysis on the recursive process and the computational savings was presented by Iowa in the late 1980s (published 1991) (7). As a result, both modeling times and computer simulation runtimes could be greatly reduced. A lead researcher for this recursive dynamics algorithm eventually left the University of Iowa and co-founded a company called FunctionBay Inc., which developed RecurDyn (8).

TARDEC scientists and engineers have been monitoring and experimenting with RecurDyn for several years to determine if the program was robust and capable. As the program matured, CASSI engineers deemed the program ready. A true pilot run was launched summer 2011. This was the first customer project to be addressed using RecurDyn at TARDEC. The following section details the basic steps in performing a vehicle dynamics analysis with RecurDyn. Specific workarounds for specific issues are not included in this paper, as there is no value added to the methodology as presented. While there is little reason to rebuild legacy track vehicle models into RecurDyn at this time, RecurDyn may be the software package of choice for future track vehicle projects.

4. Approach and Methodology

RecurDyn has a main model-building interface, solver, and a number of capabilities including toolkits for engines, tracks, belts, bearings, gears, control, hydraulics, tire interface, meshing (for FEA), pistons, valves, and a variety of others (9). RecurDyn also implements the Bekker soil model. The scope of this customer project was to measure the dynamic responses of components in a tracked vehicle subject to a number of maneuvers and course profiles, so the CoLink toolkit and the Track toolkit (a.k.a. Trackbuilder) were needed this effort. CoLink provides capability similar to Simulink and is used to provide a computer driver to the model. PID (proportional, integral, derivative) control can be implemented fairly easily with CoLink. CoLink is also fully compatible with MATLAB/Simulink(10). The Trackbuilder provides the capability to build tracks quickly and intuitively.

The model was constructed from a bottom-up approach. The track was built first, then the suspension, and finally the chassis. This allows for the lower-level elements to be simulated as they are complete, which greatly eases model troubleshooting. Note that while CAD can be imported into RecurDyn from any popular format, CAD is not necessary to run RecurDyn (note that CAD may be the best source for capturing the necessary component geometries/locations).

Once the model was completed, CoLink was used to develop the controller. This step required much effort, since heavy tracked vehicles are highly non-linear. Also, multiple commands were desired: straight path-following, double lane change maneuvers, side slopes, and steady-state turning. Proportional – Integral – Derivative (PID) controllers were used for both speed and steer control. The gains were adjusted iteratively to have a single solution for each desired maneuver. Also, a simple vehicle powertrain model was incorporated into CoLink to provide realistic power limits.

Of special note, CoLink and RecurDyn are integrated to form a single co-simulation effort (9). Whenever a virtual driver is desired, CoLink must be run to drive the vehicle. For each time step within a controlled simulation, RecurDyn feeds CoLink the desired inputs (error term, speed, direction, etc), CoLink performs the programmed operation (generates torque inputs at the sprocket), CoLink outputs the results, and then RecurDyn applies the relevant command for the next time step. Highly complex models are simulated in this matter. A full vehicle dynamic model is not needed within CoLink itself to run this co-simulation.

4.1 Modeling in RecurDyn

Modeling in RecurDyn offers several advantages over other Multi-Body Dynamics (MBD) modeling programs. Of particular note is the use of subassemblies within the model. The subassemblies give a unique advantage in that certain parts of the model can be imported/exported easily, copied and pasted easily, and can even be simulated independently from other model assemblies or subassemblies. Joints can freely attach between the different assembly levels. A best practice for building the models is to separate the functions into different subassemblies, so that each function can be simulated independently for ease of troubleshooting and validation. For example, in tractor-trailer combinations, the trailer can be its own subassembly, with sub-subassemblies for the suspension components. For the purpose of a traditional tracked vehicle, each of the driver's and passenger's side tracks will be separate subassemblies, with the frame or hull being the parent assembly.

4.1.1 Track Subassembly

RecurDyn's Trackbuilder is a user-friendly GUI which helps build the tracks. The dimensions, masses, and geometries of the sprocket, idler, idler wheels, and road wheels are all entered at this level. A characteristic track segment is also entered at this level, complete with mass, inertia, and dimensional data. Once the characteristic track segment is entered, the track assembly is built by selecting the track path around the sprocket, road/idler wheels, and idler.

The suspension should also be part of this track subassembly. While the suspension could be part of the parent assembly, having the suspension as part of the track assembly allows for quicker troubleshooting if errors arise. Since the suspension is part of the track subassembly, it can be simulated separately from the entire vehicle model. This also decreases the complexity of modeling the parent assembly.

4.1.2 Model "Parent" Assembly

The parent assembly can be as simple or as complex as is necessary, but if any part of the parent assembly is dynamically complex, the user should consider forming a separate subassembly for that function. Once the subassemblies are all finished and operating correctly in independent simulations, the parent assembly should then be used for overall integration. Once the subassemblies have been imported and integrated correctly at the parent level, the model is ready for a basic simulation run.

4.1.3 Basic Simulation Run

Once the subassemblies are integrated successfully into the parent assembly, a simple simulation (such as velocity input at the track sprockets) should provide a simple test to ensure the model is functioning correctly. This can be done quickly by building a large, flat road and editing the properties of the sprocket revolute joints to include motion. The user should then run a simulation and both view the simulation animation and load the results into the "Plot" post-processor within RecurDyn. The user should ensure that the forces and motions appear realistic and reasonable, and the user should troubleshoot any apparent problems. Once the model is satisfactory, then the advanced controller should be built.

4.2 Control in RecurDyn

CoLink provides control functionality to the RecurDyn suite and is equipped with many signal generating and processing capabilities (10). CoLink combines with the RecurDyn solver to perform a co-simulation. For each time step in the co-simulation, RecurDyn and CoLink communicate. For instance, RecurDyn passes CoLink an error position term, error heading term, and a speed term; and CoLink analyzes this data to generate the proper torque commands of both the driver's and passenger's sprockets to pass back to RecurDyn. For the next time step, the new torques are applied, the solver analyzes the reactions of the vehicle model, and generates the applicable outputs once again and passes them to CoLink. While the native capabilities of Colink are numerous, the package offers even more capability since it can be linked to MATLAB/Simulink for more advanced computational methods.

4.2.1 Setting up the Model for Control

RecurDyn passes output values to CoLink through a "Plant Output" tool. This plant output tool can send any number of data values to CoLink. Data may include user-defined constants (such as overall vehicle length and width, which is used for the NATO double lane change maneuver) or may consist of simulation-derived values (axial position, velocity, acceleration, speed, distance, force, torque, radial velocities, literally thousands of possibilities). For the purpose of this analysis, the following were plant outputs:

- Look-Ahead Point – X (longitudinal) and Z (Lateral) positions and velocities (Look-Ahead Point is aligned vertically and laterally with vehicle center of gravity and is longitudinally ahead of vehicle by a factor of about 2x vehicle length).
- Turning-circle course focal point X and Z position
- Look Ahead Point's scalar speed and acceleration
- Vehicle Center of Gravity (CG) X and Z position

The look-ahead point is essential for proper steer control. Figure i below shows a basic schematic of the look-ahead point.

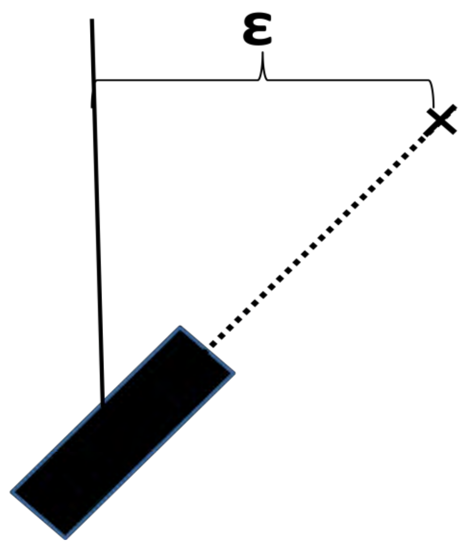

Figure i - Vehicle Look-Ahead Point with Proportional Error Value Shown

The look-ahead point serves as a combination of a future predictor and heading indicator. Since the look-ahead point is directly in front of the vehicle, it predicts the vehicle's future location if there was no steer command given. The look-ahead point is beneficial to steer control. In Figure i above, the vehicle's center of gravity is centered with the desired path. If a look-ahead point was not used, the vehicle would go well off track. With the look-ahead point active, the controller is able to adjust the steer command immediately to return the vehicle to the correct orientation and path.

Additionally, several constants may be set as plant outputs in RecurDyn or, alternatively, can be made into "Constant" blocks within CoLink (user's preference). These necessary constants include

- Vehicle length
- Vehicle width
- Max braking torque available

Note that the max torque available at the sprocket needs to be calculated as well. This is a function of the engine's torque map, the vehicle speed, the transmission, and the final drive ratio; and is better handled within CoLink.

Once these plant outputs are established, the "Plant inputs" need to be defined. The plant inputs are the values that CoLink feeds back to RecurDyn. For the purposes of this analysis, the following plant inputs were used:

- Driver's side sprocket drive torque
- Passenger's side sprocket drive torque

CoLink can now be initialized and connected.

4.2.2 Building the CoLink Modules

The model used for this analysis is a three-way controller with one Proportional-Integral-Derivative (PID) controller for speed, one PID controller for steering, and one algorithm to calculate the maximum available vehicle torque. The four figures below highlight this control strategy. Several key assumptions for this controller:

- The "clock" and "switch" blocks allow for a given amount of time (1 second) before any error is accumulated and before any command is given back to RecurDyn. This allows for the model to settle within the simulation prior to any command.
- The simple engine torque model used is not ideal, but for this analysis a limited amount of data was available. The lookup table used gives the max available torque output from the transmission (max engine torque x gear ratio of lowest gear possible at given speed) for a given vehicle speed.
- The "max" and "min" blocks maintain a realistic torque command, both for positive (engine effort) and negative (brake effort).
- "Rate limiter' blocks were used to filter any high frequency command torques.

In the actual controller file built, there are separate sub-assembly blocks for each maneuver type. Each sub-assembly block needs to be tuned (PID) separately. When the user wants a certain maneuver, the user only needs to change which signal is input into the final assembly module.

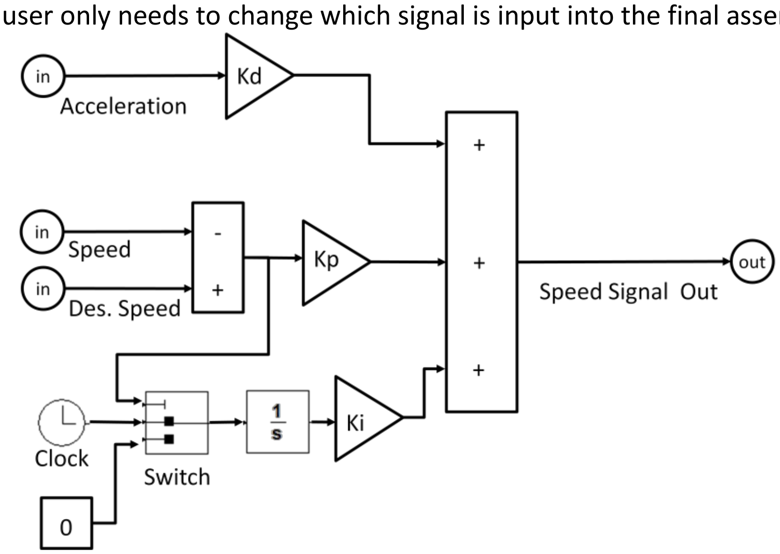

Figure ii - Speed Control

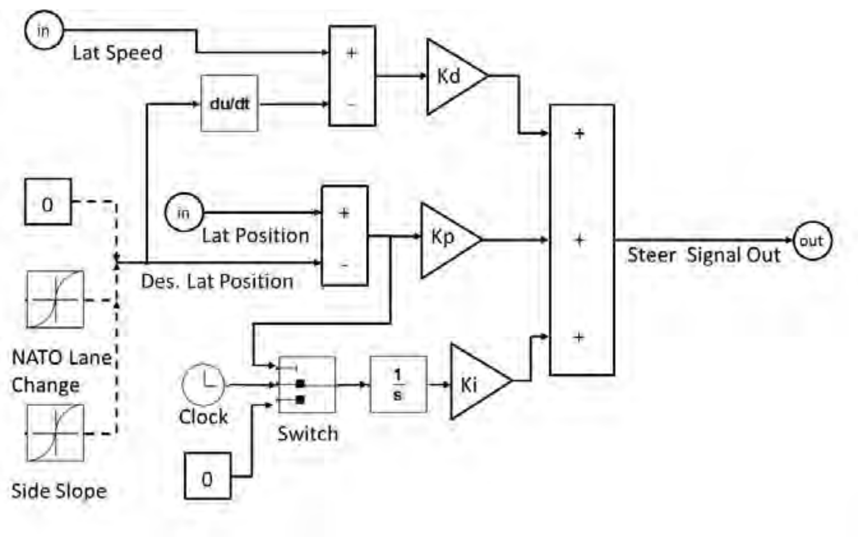

Figure iii - Representative Steer Model

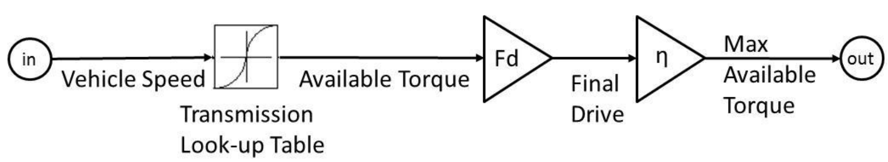

Figure iv – Representative Max Available Torque

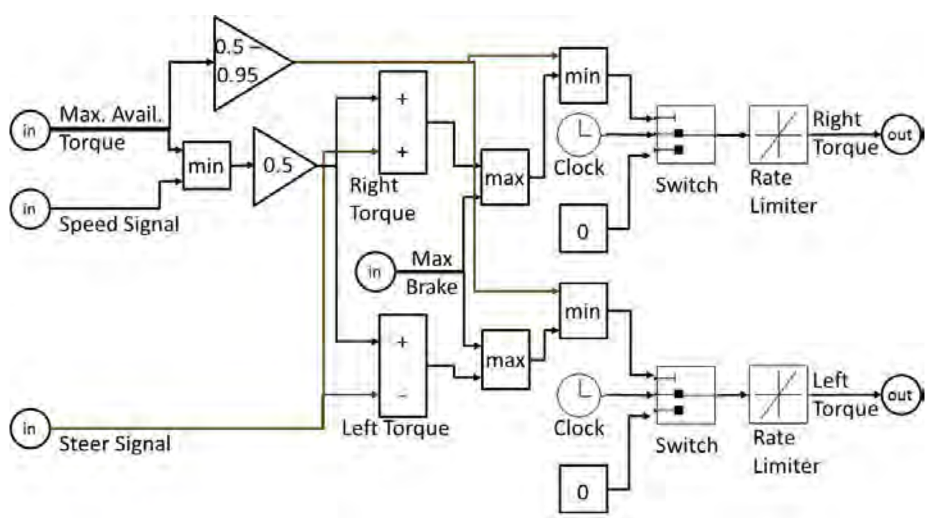

Figure v - Representative Final Module

The steady-state turning circle test utilized a slightly different steer method than Figure iii above. The steady-state turning circle steering was controlled as shown in Figure vi below (X and Z values below are axial positions; "Actual Distance" is the real-time distance between the center of the turning circle and the vehicle; "_aim" refers to the look-ahead point).

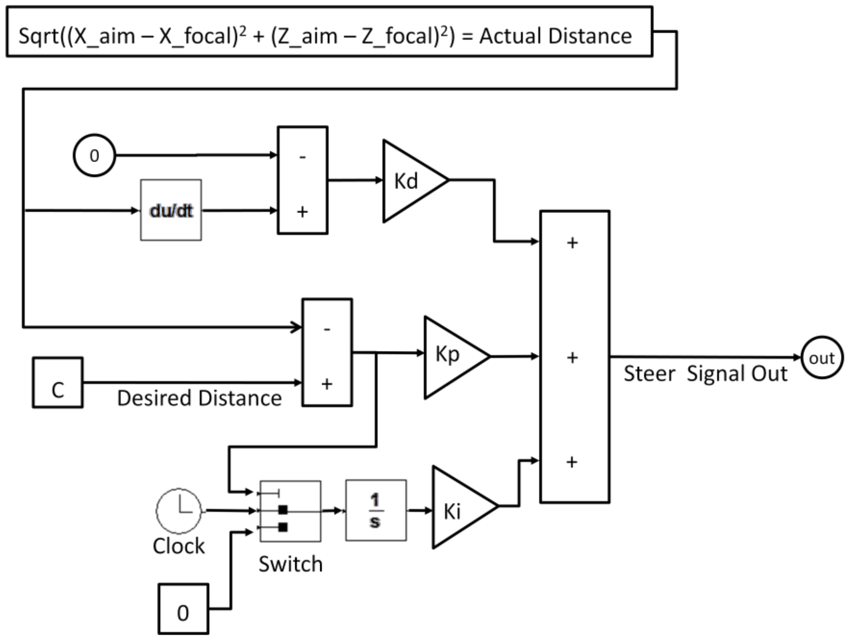

Figure vi - Steady State Turning Circle Steer Command

On a final note for analyzing the success of the controller, the vehicle CG positions, and not just the look-ahead coordinates, should be monitored by the user to ensure that the controller is effective.

4.3 Simulating the Model

Once the model and controller have been built, simulations can begin. For each desired steering maneuver, several iterations will be needed to tune the PID controller. There are different methods available for tuning the controller, and a very useful tutorial is available at the following web reference (11). If needed, both CoLink and RecurDyn have output-to-file and export capabilities to analyze data. Load histories at certain points can also be generated through RecurDyn simulations and sent for further analyses (such as FEA).

5. Results, Process Validation, and Discussion

During the course of this analysis, two separate vehicles were modeled and simulated. During the course of the first model, there was a large learning curve and resulted in many lessons learned. While there are tutorials and manuals available for RecurDyn, many specific modeling issues were present that weren't specifically addressed and slowed the initial modeling process. Through advice from expert M&S engineers both within and outside of TARDEC, the model and controller were successfully completed. For the second vehicle, the modeling process was much quicker, and the only significant effort was retuning the controller. The lessons learned have been documented to assist future users.

The controller requires tuning for each vehicle (and vehicle configuration) tested. This is a cumbersome process that can take anywhere between a few hours to a few days per maneuver, depending on the model simulation time (depending on the level of complication in the model). A possible solution to this issue is to build an adaptive controller, with preview capability.

6. Conclusions and Recommendations

Ultimately, the RecurDyn package is very useful in generating timely analyses for TARDEC customers. While there were several issues that arose, the issues were overcome and work-arounds now exist. RecurDyn is able to efficiently solve complex interactions without any loss of data fidelity. The templates and controller that were generated will certainly evolve over time. This is to be expected, as different users or different models may discover further issues that need to be resolved. The path forward for improving modeling times is to develop an adaptive control, with preview capability, which will eliminate the need to re-tune the controller for each simulation course and vehicle.

7. Works Cited

1. *A RECURSIVE IMPLEMENTATION METHOD WITH IMPLICIT INTEGRATOR FOR MULTIBODY DYNAMICS.* **H. J. Cho, H. S. Ryu, D. S. Bae, J. H. Choi, B. Ross.** Pittsburgh, PA : American Society of Mechanical Engineers, 2001. DETC. DETC2001 / VIB-21319.

2. *A Generalized Recursive Formulation for Constrained Mechanical System Dynamics.* **D. S. Bae, J. M. Han, H. H. Yoo.** 3, s.l. : Marcel Dekker Inc, 1999, Mechanics of Structures and Machines, Vol. 27. 0890-5452, 1525-612X.

3. *A compliant track link model for high-speed, high-mobility tracked vehicles.* **H. S. Ryu, D. S. Bae, J. H. Choi, A. A. Shabana.** 10, s.l. : John Wiley & Sons, Ltd, June 2000, International Journal for Numerical Methods in Engineering, Vol. 48. 10.1002/1097-0207(20000810)48:10<1481::AID-NME959>3.0.CO;2-P.

4. **James Lacombe, Mark Moran, Scott Happel.** *A 3-Dimensional Dynamics Model for Generating Tracked Vehicle Seismic Signals.* s.l. : US Army Corps of Engineers, 2000.

5. *Dynamics of High Mobility Track Vehicles.* **M. K. McCullough, E. J. Haug.** 2, Iowa City, IA : ASME, June 1986, Journal of Mechanisms, Transmissions and Automation in Design, Vol. 108. Department of Mechanical Engineering and Center for Computer Aided Design, College of Engineering, The University of Iowa. 10.1115/1.3260801.

6. *A Recursive Formulation for Real-Time Dynamic Simulation of Mechanical Systems.* **Dae-Sung Bae, Ruoh-Shih Hwang, Edward J. Jaug.** 2, Iowa City, IA : American Society of Mechanical Engineers, June 1991, Journal of Mechanical Design, Vol. 113, p. 9. Center for Simulation and Design, Optimization of Mechanical Systems and Department of Mechanical Engineering, College of Engineering, The University of Iowa. 10.1115/1.2912764 .

7. *A Decoupled Flexible Relative Co-ordinate Recursive Approach for Flexible Multibody Dynamics.* **Hao-Jan Lai, Edward J. Haug, Sung-Soo Kim, Dae-Sung Bae.** 8, Iowa City, IA : John Wiley & Sons LTD, December 1991, International Journal for Numerical Methods in Engineering, Vol. 32. Center for Simulation and Design Optimization of Mechanical Systems, The University of Iowa. 0029-5981, 1097-0207.

8. **FunctionBay Inc.** RecurDyn - Company. *RecurDyn - Multi-Body Simulation Software.* [Online] 2011. http://www.recurdyn.de/company.php.

9. **MotionPort, LLC.** Recurdyn Brochure. *MotionPort.* [Online] http://www.motionport.com/Brochures/RecurDyn.PDF.

10. **FunctionBay, Inc.** *User's Guide for RecurDyn.* s.l. : FunctionBay, Inc, 2010.

11. CTMS: PID Tutorial. *Control Tutorials for MATLAB and Simulink.* [Online] William Messner and Dawn Tilbury, 1998. http://www.engin.umich.edu/class/ctms/pid/pid.htm.

12. *An Optimal Preview Control for Linear Systems.* **MacAdam, Charles C.** s.l. : ASME, September 1980, Journal of Dynamic Systems, Measurement and Control, Vol. 102, pp. 188-190.

13. **MacAdam, Charles C.** *Development of Driver/Vehicle Steering Interaction Models for Dynamic Analysis.* Ann Arbor, MI : The University of Michigan Transportation Research Institute, 1988. For The U.S. Army Tank Automotive Command. UMTRI-88-53.

14. *Application of an Optimal Preview Control for Simulation of Closed-Loop Automobile Driving.* **MacAdam, Charles C.** No 6, s.l. : IEEE, 1981, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 11, pp. 393-399.

15. *An Adaptive Lateral Preview Driver Model.* **Ali Y. Ungoren, Huei Peng.** 4, s.l. : Taylor & Francis Group Ltd, April 2005, Vehicle System Dynamics, Vol. 43, pp. 245-259. 0042-3114 / 1744-5159.

16. *Approximating Kinematics for Tracked Mobile Robots.* **J. L. Martinez, A. Mandow, J. Morales, S. Pedraza, A. Garcia-Cerezo.** 10, s.l. : Sage Publications, October 2005, International Journal of Robotics Research, Vol. 24, pp. 867-878.

17. *Design of Turning Control for a Tracked Vehicle.* **Geng G. Wang, Shih H. Wang, Cheng W. Chen.** s.l. : IEEE, 1990, Control Systems Magazine.

18. *Look-ahead preview control application to the high-mobility tracked vehicle model with trailling arms.* **Okhyun Kang, Youngjin Park, Youn-sik Park, Moonsuk Suh.** s.l. : KSME & Springer, 2009, Journal of Mechanical Science and Technology, Vol. 23, pp. 914-917.

19. *Vehicle Dynamics Applications of Optimal Control Theory.* **R. S. Sharp, Huei Peng.** 7, s.l. : Taylor & Francis Group, July 2011, Vehicle Systems Dynamics, Vol. 49, pp. 1073-1111. ISSN 0042-3114 print/ISSN 1744-5159 online.