Test and Evaluation/Science and Technology Program

Net-Centric Systems Test Focus Area

PREdictive Smart Synchronization (PRESS) Algorithms;

Removing Data Bias from the Virtual Test Range

Farrokh Vatan[1], Anthony Barrett[1], Ed Chow[1], Mark James[1],
Kent Pickett[2], and Gil Torres[3], Jason Allen[3]

[1] NASA's Jet Propulsion Laboratory, California Institute of Technology
[2] MITRE
[3]Naval Air Weapons Division

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

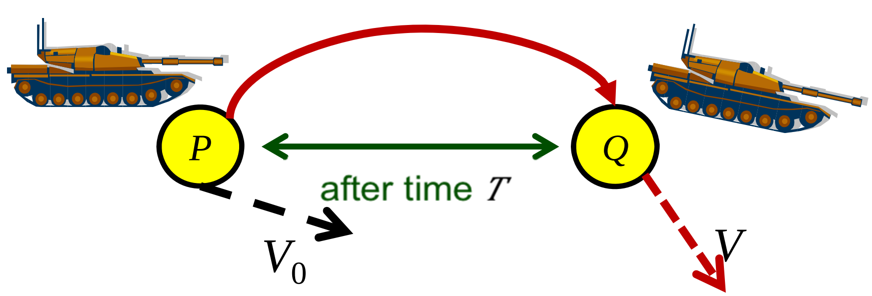
Acknowledgement

This project is funded by the Test Resource Management Center (TRMC) Test and Evaluation/Science & Technology (T&E/S&T) Program through the U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI) under Contract No. NAS7-03001, Task Plan Number 81-12346. The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology. Technologies developed support Naval Air Weapons Station, China Lake.

Introduction

- Dead Reckoning (DR) algorithms have been used in Distributed Interactive Simulations (DIS) for over 35 years
- A node uses the received Time Space Position Information (TSPI) to put the other node at position P and begins moving the object at velocity V_0 with acceleration A_0 and then the DR position at a specific time T is calculated
- The DR algorithm is also used in live testing to reduce the TSPI synchronization error caused by bias from the distributed test environment

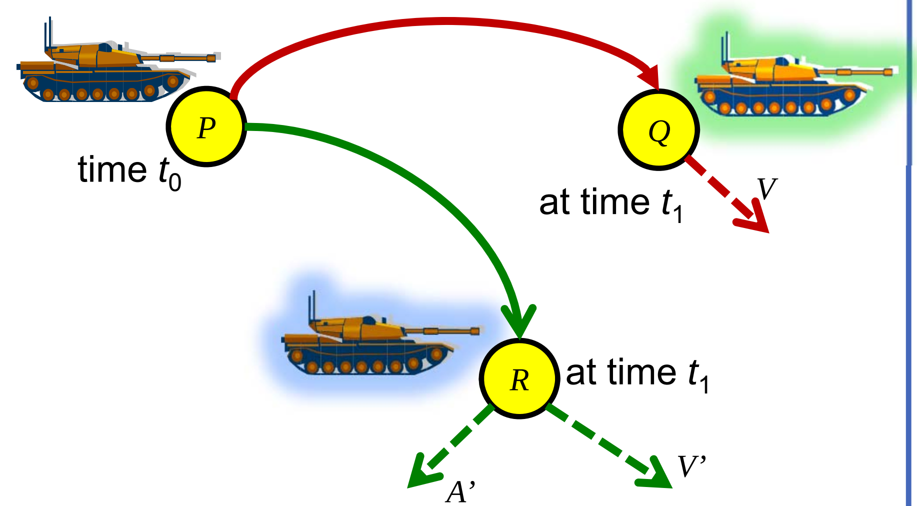
Dead-Reckoning Calculation



$$Q = P + V_0T + \frac{1}{2}A_0T^2$$

$$V = V_0 + A_0T$$

Dead-Reckoning Synchronization Error



Standard Dead Reckoning Algorithms

Dead Reckoning Models

Field	Model	Formula
1	STATIC	N/A
2	DRM (FPW)	$P = P_o + V_o \Delta t$
3	DRM (RPW)	1) $P = P_o + V_o \Delta t$ 2) $[R]_{w->b} = [DR][R_o]_{w->b}$
4	DRM (RVW)	1) $P = P_o + V_o \Delta t + \frac{1}{2} A_o \Delta t^2$ 2) $[R]_{w->b} = [DR][R_o]_{w->b}$
5	DRM (FVW)	1) $P = P_o + V_o \Delta t + \frac{1}{2} A_o \Delta t^2$
6	DRM (FPB)	1) $P = P_o + [R]^{-1} w->b(R1V_b)$
7	DRM (RPB)	1) $P = P_o + [R]^{-1} w->b(R1V_b)$ 2) $[R]_{w->b} = [DR][R_o]_{w->b}$
8	DRM (RVB)	1) $P = P_o + [R]^{-1}_{w->b}(R1V_b + R2A_b)$ 2) $[R]_{w->b} = [DR][R_o]_{w->b}$
9	DRM (FVB)	1) $P = P_o + [R]^{-1}_{w->b}(R1V_b + R2A_b)$

Source: IEEE Standard 1278.1-1995

State of the Art: Extended DR

There are several suggestions to improve performance of the standard dead reckoning algorithm:

- Map-based Dead-reckoning: Used for local positioning in urban areas
  - *A. Leonhardi, C. Nicu, and K. Rothermel, A Map-based Dead-reckoning Protocol for Updating Location Information, 2001*
- DR algorithm based on artificial potential field: Used to avoid obstacles in gaming simulation
  - *X. Shi, X. Wang, J. Bi, F. Liu, D. Yang, and X. Liu, A DR algorithm based on artificial potential field method, 2009*
- Dead reckoning algorithm based on kinematic model analysis: Used for mobile robots to build its kinematic model in complex terrain and to estimate its dead reckoning based on that kinematic model
  - *J. YU, Z. Cai, and Z. Duan, Dead reckoning of mobile robot in complex terrain based on proprioceptive sensors, 2008*
- Position history-based DR: The algorithm uses either a second-order (parabolic) estimation between the three most recent updates or a first-order (linear) estimation between the two most recent position updates
  - *S. Singhal, Effective Remote Modeling in Large-Scale Distributed Simulation and Visualization Environments, 1996*
- Adaptive adjustment of threshold level: A multi-level threshold scheme is proposed to handle the dynamic relationships between moving entities
  - *B. Lee, A. Cai, S. Turner, and L. Chen, Dead reckoning algorithm for distributed interactive simulation, 1999*

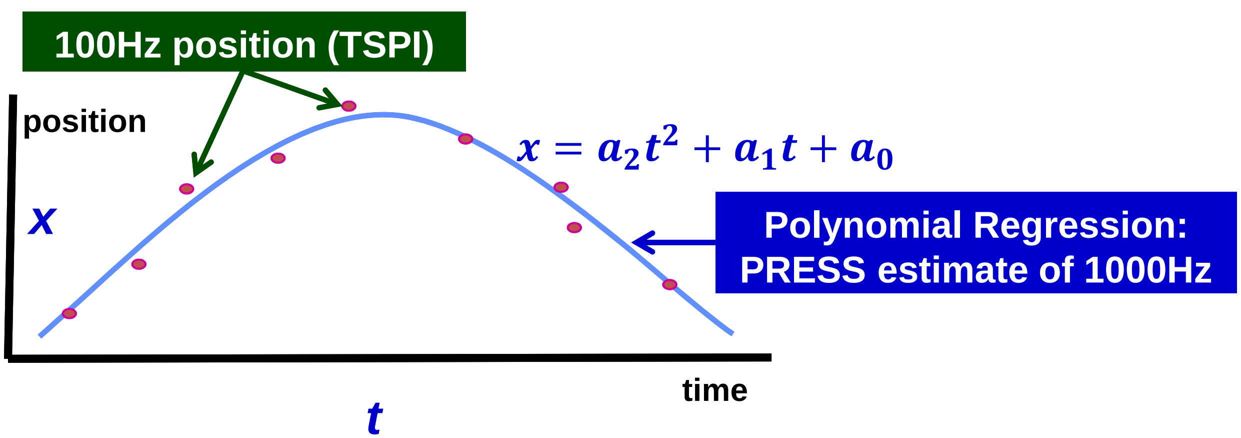
Observations

- Standard DR algorithms use a constant acceleration value during each TSPI update period. Because of this constant acceleration assumption in the standard DR algorithms, DR-based synchronization is prone to error when dealing with signals that vibrate rapidly with respect to the latency period.

- Due to a combination of noisy sensor(s) on the System Under Test (SUT) and network conditions, standard DR algorithms can result in high errors for some SUTs under certain test conditions

PREdictive Smart Synchronization (PRESS) Algorithms

- The PRESS algorithm continuously monitors model correctness and computes and distributes new models as necessary
  - Curve fitting is based on the previous position data: through n^th order polynomial depending on fitting error and computational limitations

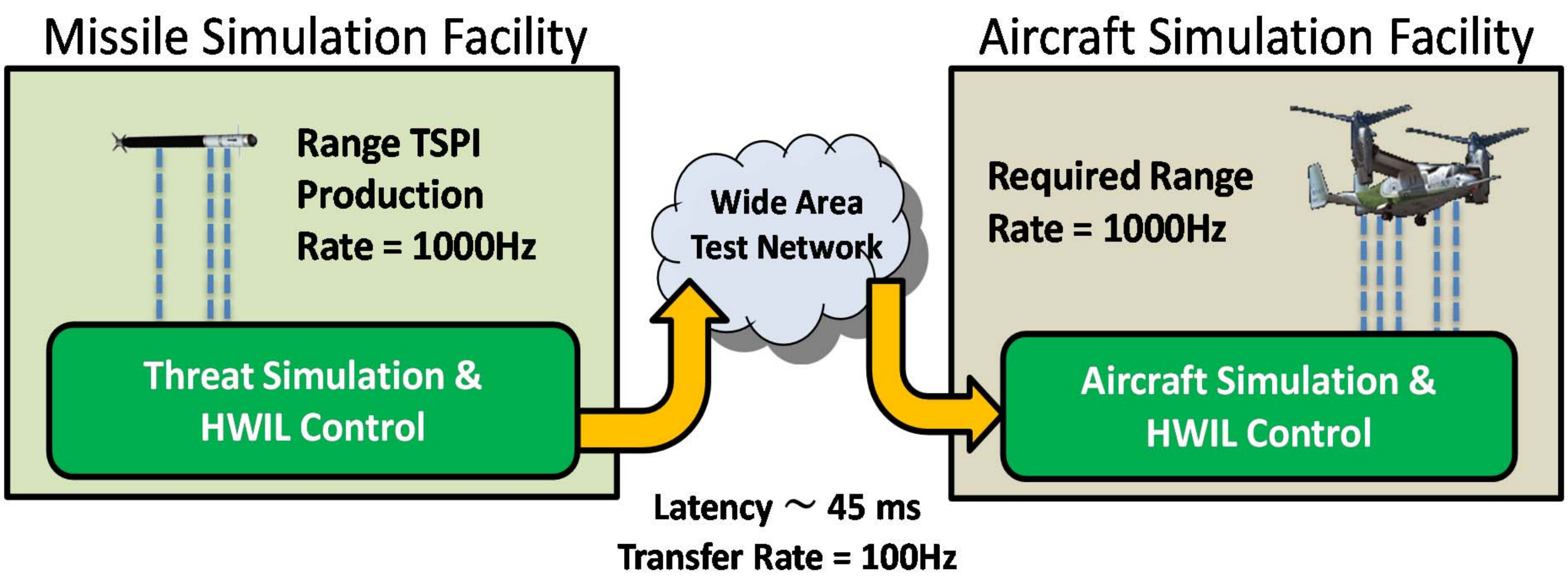


PREdictive Smart Synchronization (PRESS) Algorithms (cont'd)

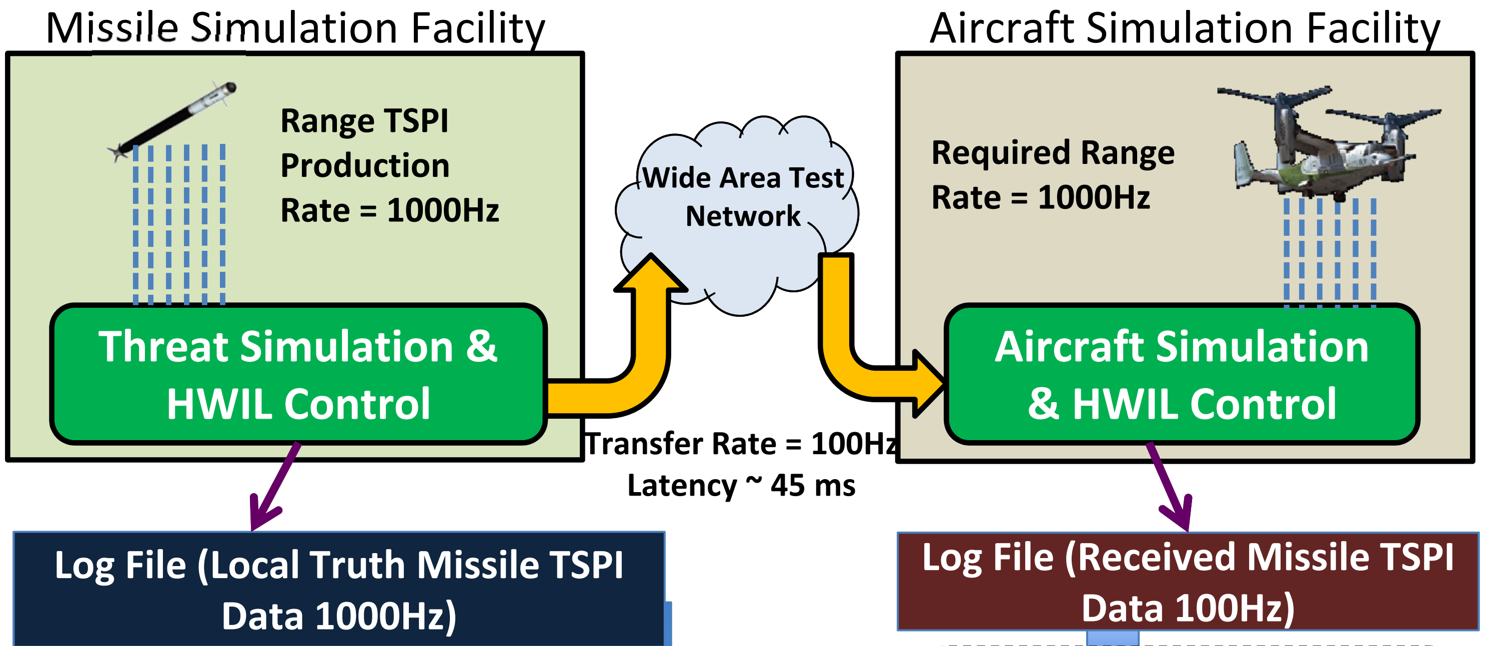
- Only use the position information to determine the acceleration constant.
  - The position information does vibrate, but fitting to the extra points reduces the vibration induced error that results in a much smaller latency derived error.
- Computes a new polynomial for each TSPI component whenever a new TSPI arrives from the network
- Computes a new TSPI from these polynomials on demand when given a synchronization time *t*

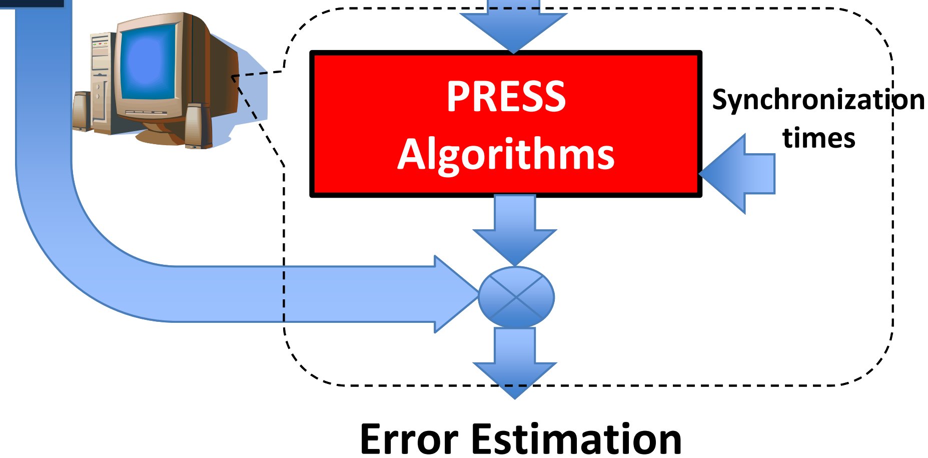
PRESS Use Case

- **Joint Distributed Infrared Countermeasures (IRCM) Ground-test System (JDIGS)**
  - Effectiveness testing of IRCM aircraft protection systems installed on U.S. military aircraft



Using JDIGS Data to Evaluate PRESS Algorithm Performance

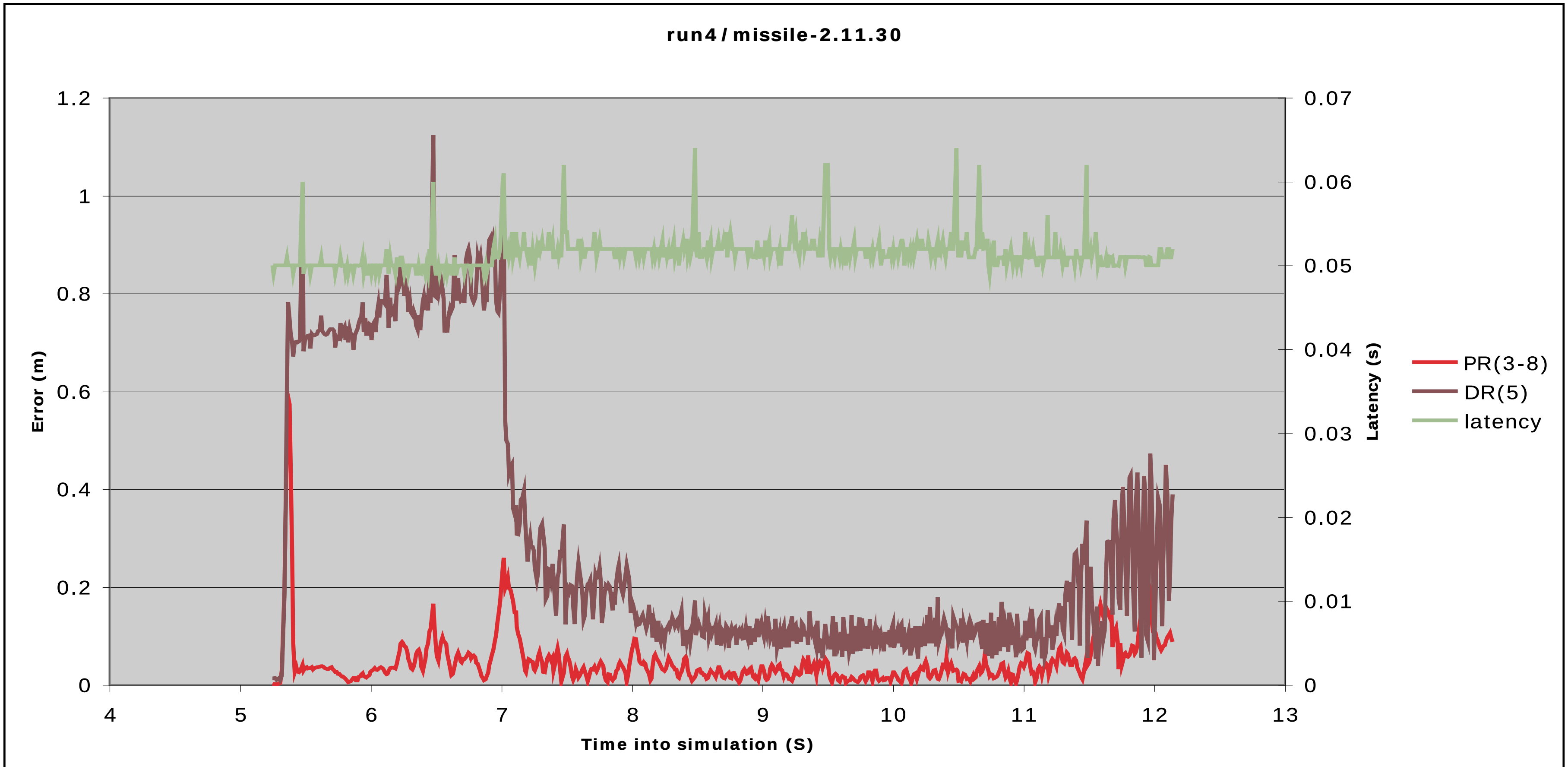




- **Missile TSPI data production at 1000Hz**
- **Network allowable transfer data rate at 100Hz**
- **Network latency is approximately 45 ms**
- **PRESS algorithms “synchronize” truth and simulated missile TSPI data:**
  - **Removes network latency bias and error caused by packet drop**
  - **Predicts intermediate data at required range rate**

PRESS Performance Improvement

- **PRESS algorithm achieved over 80% improvement on overall position error performance compared to standard DR algorithm**



DISTRIBUTION STATEMENT A

Summary

- We developed a new class of algorithms to synchronize TSPI in a distributed live testing environment

- Our PRESS algorithms uses real-time dynamic modeling approach to reduce the bias from distributed testing environment

- Our simulation indicated significant error performance improvement over standard DR algorithms for the JDIGS use case and other use cases.

Points of Contact

Mr. Gil Torres
Net-Centric System Test Executing Agent
Naval Air Weapons Division, Pt. Mugu
gilbert.a.torres@navy.mil

Ms. Kathy Smith
Net-Centric System Test Deputy Executing Agent
ksmith@gblsys.com

Mr. Kent Pickett
NST Subject Matter Expert
kpickett@mitre.org

Dr. Edward Chow
Principal Investigator (PI)
edward.t.chow@jpl.nasa.gov

Mr. Jason Allen
Naval Air Weapons Division, China Lake
jason.r.allen@navy.mil