Digital Butterfly as Research and Education Target

TAKAO FUTAGAMI[†1] CORTLAND STARRETT[†2]

Abstract: IoT environments open new application fields where traditional digital designers have been afraid to enter. Challenges such as accurate computation of mathematical integration will become possible using heavy approximation software execution. These environments also give opportunities to students to learn to cope with computational error canceling problems.

Keywords: IoT, Digital Butterfly, approximation software, IMU, gait analysis

1. Introduction

IoT environments will open new application fields where traditional digital designers have hesitated to address such as accurate computation of mathematical integration becoming possible using heavy approximation software execution. These environments also provide a good opportunity for students to learn how to cope with computational error canceling problems.

This paper focuses attention on a problem not yet solved in which robotics applications need IoT environments for practical use. It is named the *Digital Butterfly Problem* and shall be targeted as both a research and an education issue.

1.1 APRIS research and education target

APRIS (spelled out here) is focusing on physical object control and management systems such as robot control and automotive applications though embedded system can be applied to non-physical problems as well. One clear motivation for focusing on physical targets is the high demand from industry. Upon graduation, students can get jobs instantaneously. However, another benefit is also important for students; they can easily discern if their applied learning is correct, because it is clearly reflected in physical target behavior

Consider a smart phone embedded recommendation application using GPS and magnetic direction sensors. If the developer runs a system test and requests a recommendation of a good *dumpling* restaurant, he expects a good Chinese restaurant name. However, did he fail in his development if the system gives one Chinese restaurant and one Italian restaurant? It is not obvious, because the system might recommend a good *ravioli* restaurant.

1.2 Physical robot is fail obvious application.

Then, suppose a student team is asked to develop a robotic car system and has completed the first prototype. When the robot car run autonomously for the first time, it runs out of control and crashes into the rail. In this case, it is immediately clear that student understood something wrong.

Any physical application development or embedded system applying the laws of dynamics is highly useful in industry and promotes self-learning.

2. Butterflies in real and digital world

2.1 Real Butterfly i.e. butterfly effect

When we introduce digital application development as an education examples, we should understand an important issue called the *butterfly effect* when dealing with dynamics problems on computer system. Any embedded MPU, DSP, FPGA, or GPU is a good platform to develop dynamics controllers and solvers. Note that there are two butterflies. One is Real butterfly, and another is Digital Butterfly.

2.2 Real Butterfly i.e. butterfly effect

Real butterfly refers to the well known "butterfly effect" in physical dynamics. When students learn the fundamentals of dynamics in high school, they understand that all physical objects (mass or rigid body) follow Newton's law. Such physical object behavior is computable so far as the initial conditions and constraints are defined. At the university level, students learn that computer software allows people to predict complex astronomical dynamics even though such problems are impossible to solve analytically.

A big disappointment comes after this simple and beautiful understanding.

Students learn that a very small initial condition difference causes large differences in physical world outcomes. A small difference such as is caused when one butterfly flapping its wings in Thailand might be the cause of a big typhoon. This butterfly fable is well accepted to represent the butterfly effect in general.

2.3 Digital Butterfly

In the digital computation world, we unfortunately also have a digital butterfly. Some parts of the digital butterfly effect are well known and understood by electronics engineers. This effect includes many causes such as:

- A/D converter quantization problems are well known. It is not an error but is a systematic design issue. Nobody knows if the real value is 2.33 or 2.56, because the A/D convertor output is always a digital approximation. Without good numeric compensation, dynamic path computation introduces errors so large that the technique is rendered useless.
- Sensor output randomness causes random walk problems when analysis software receives digital values from the sensor signal processing circuit.
- Electronics systems between the sensor and the software create so much noise and bias error that the software has a hard time to compute appropriate output.

The problem is like the butterfly effect in dynamics. A small

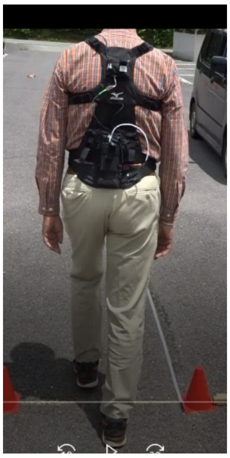
†1 Toyo Corporation.
†2 One Fact inc.

error or change of initial conditions or the reading of sensor values causes vastly different computational results.

2.4 Example

To understand one practical problem, let us consider human gait (walking) analysis.

Precisely measuring a human's walking stride provides excellent predictive data for medical research of healthcare. The author is involved in this medical research as data acquisition and path analysis specialist. The PUT (Person Under Test) stands at the starting line wearing an IMU (Inertial Measurement Unit) data recorder. The PUT is asked to walk naturally in a straight line for a distance of 15 meters (left Photo1). Recorded data is analyzed by software. The base software functionality is to compute initial IMU angle and walking velocity through acceleration sensor data and angular velocity sensor data.



$$\vec{V}(\mathrm{t})=\vec{V}(\mathrm{t_0})+\int_{t0}^{tend}\vec{\alpha}\,(t)dt$$

The next step is then computing the walking path from integral of $\vec{V}(\mathrm{t})$.

$$\vec{P}(\mathrm{t})=\vec{P}(\mathrm{t_0})+\int_{t0}^{tend}\vec{V}\,(t)dt$$

2.5 Before Cancelation

Fig1. Walking path
omega ofst=-0.0030

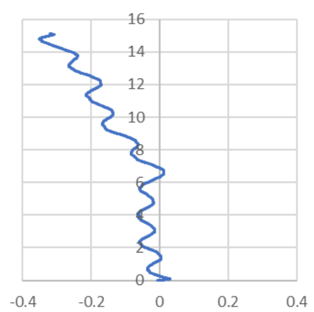
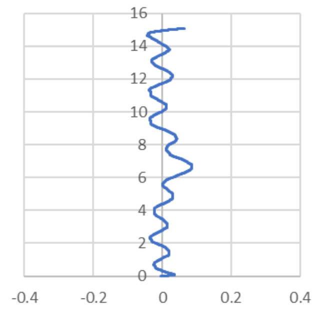


Fig1 and Fig 2 are example of analysis. These figures show the walking path of the PUT as viewed from above. The input for the analysis software is taken while PUT is walking 15m. Both used exactly the same IMU sensor data set (3 linear acceleration sensor signals and 3 angular velocity sensor signals) and the same experimental conditions (distance walked). Fig.1 is Digital butterfly affected path graph. It appears as though the PUT made left turn while walking though PUT actually made no turns.

2.6 After Cancelation

Fig.2 is Digital butterfly canceler applied path. It accurately reflects the true walk path of the PUT. What is noteworthy here is that canceling the digital butterfly effect is done by adding a small correction factor. In this case it was 0.005 degree/second rotation speed in the z-axis. In this system, the rotation speed resolution of the sensor is 0.02 degree/second for this walking analysis because of allowed weight limit. 0.005 degree/sec is 1/20 of sensor resolution and is not measurable. A tiny number difference creates this dramatic change. This small digital butterfly cancelation value is computed by a heavy approximation iteration loop. At this moment it takes half an hour to calculate on a modern personal computer. It does not use Kalman filter type model at all because of the required accuracy and difficulty of human dynamic model development.

Fig2. Walking path
omega ofst=0.0020



2.7 Traditional discipline

This measurement and analysis system was initially designed in 2010 and took 3 years to complete the Digital Butterfly Canceling algorithm. It is not universal and has limited scope. Also, it takes such a long time to get path output that modern IoT technology should be applied to make the process more efficient.

Weather forecasting systems might use super computers when predicting a typhoon path to compensate for this butterfly effect, but most embedded applications cannot be equipped with super computer. Therefore, up until recently, digital engineering discipline advises, "Do not apply embedded digital solution to computationally heavy problems."

3. IoT is key to deal with Digital Butterfly

Attendees of this symposium understand that IoT will enable us to penetrate this barrier in many applications with high speed networks, cloud data repositories, and computation service environments. Therefore, learning to cope with each Digital Butterfly problem is an important focus for research and development to improve human health, country infrastructure, global business relationship and more. However, a challenge is that each type of Digital Butterfly depends upon the characteristics of each application and no single simplistic algorithm or solution exists. Also, students must understand such problem while they are in school and prepare to join the professional IoT development world.

Any question is welcome to takao@toyo.co.jp

Reference

[1]The 25th Annual Meeting of Japanese Society of Biomechanics
Sports science contribution to Pincham Olympic game
Yuki, Kodaira, Futagami, Aoki

[2]IEICE Technical Report
An Experiment of Supporting Self Reflection based on the Visualization of Characteristics of Velocity in the Speed Skating
Yu Takahashi, Kayama, Hashimoto, Futagami, Yuki

Acknowledgments

This system development has done with Tsukuba university, Shinshu university, FCT and SESSAME members.

Yu Takahashi was the major contributor of programing this canceller algorism.