

Опанасенко В.Н., д. т. н.,
orcid.org/ 0000-0002-5175-9522,
Завьялов С.Б., к. т. н.,
Софіюк О.Т.

АППАРАТНОЕ ЯДРО НА БАЗЕ FPGA ДЛЯ РОБАСТНЫХ АЛГОРИТМОВ ОЦЕНКИ ВЕКТОРА СОСТОЯНИЯ И УПРАВЛЕНИЯ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

Институт кибернетики им. В.М. Глушкова НАН Украины

opanasenkoincyb@gmail.com
radionix13@gmail.com
otsof@yandex.ua

Введение

Функциональная полнота задач, которые способны решать современные и перспективные малые космические аппараты (МКА) во многом определяются точностью их систем навигации и управления ориентацией. Создание этих систем, отвечающих все более ужесточающимся требованиям, не представляется возможным без использования последних достижений теории управления динамическими системами (ДС) в условиях неопределенности [1]. Под неопределенностью тут понимается неоднозначность в информации о структуре и параметрах математических моделей ДС, их текущего вектора состояния, о свойствах неконтролируемых помех измерения и действующих внешних возмущений. Для решения задач управления в условиях неопределенности первыми стали использоваться методы, основанные на вероятностной интерпретации неопределенности. К этим методам относятся широко известные алгоритмы фильтра Калмана [2], которые получили широкое распространение в навигации и управлении изделиями аэрокосмической техники и других подвижных объектов. Их широкое распространение объясняется тем, что кроме фильтрации помех они позволяют по доступным неполным измерениям вектора состояния ДС получать его полные текущие оценки. Однако к недостаткам таких методов относится большой объем

априорной информации о вероятностных свойствах неопределенности, известных на этапах разработки систем управления недостаточно точно.

Начиная с конца 60-х годов прошлого столетия стали интенсивно развиваться методы, основанные на теоретико-множественной или гарантированной интерпретации неопределенности. При этом свойства неопределенных величин полностью характеризуются множествами их возможных реализаций. Они задаются гарантированными интервалами или компактными множествами своих возможных значений. В методах оценивания и управления на основе гарантированного подхода используется существенно меньше априорной информации о свойствах неопределенности, чем при вероятностном подходе. В качестве множественных оценок свойств неопределенности наряду с интервальными оценками широко применяются их определенные обобщения – выпуклые многогранники или многомерные эллипсоиды в соответствующих пространствах. Разработаны робастные методы эллипсоидального оценивания состояния ДС, сохраняющие свою работоспособность при определенных отличиях свойств неопределенности от ее априорных оценок, используемых в соответствующих алгоритмах [1]. Создание проблемно-ориентированных средств на основе чипов FPGA для систем управления

современными МКА обеспечит эффективную реализацию ряда алгоритмов, в частности, алгоритмов управления ориентацией [3-6]. Кристаллы ПЛИС фирмы Xilinx успешно использованы в системах перемещения по планете марсоходов SpiritRover и OpportunityMER [7-9].

Работа посвящена исследованию особенности реализации робастных методов гарантированного оценивания в бортовом вычислительном комплексе управления ориентацией орбитального МКА, и построенном на основе чипов FPGA.

Постановка задачи

Для решения задачи реализации метода гарантированного эллипсоидального оценивания используется робастный метод эллипсоидального оценивания. Предполагается, что задан центр \hat{x}_k и матрица эллипсоида вида $H_k = H_k^T > 0$ в момент $t = t_r$:

$$E_k = E[\hat{x}_k, H_k] = \{x \in R^n : \sigma(\hat{x}_k, H_k) \leq 1\},$$

где $\sigma(x, \hat{x}_k, H_k) = (x - \hat{x}_k)^T H_k^{-1} (x - \hat{x}_k)$.

Полагается, что неизвестный вектор $x_k^* \in E[\hat{x}_k, H_k]$. Для решения задачи используются уравнения, полученные в результате линеаризации исходных уравнений математической модели в окрестности точки-оценки \hat{x}_k .

Ставится задача определения центра-оценки \hat{x}_{k+1} и матрицы эллипсоида вида H_{k+1} , для которого гарантируется $x_{k+1}^* \in E[\hat{x}_{k+1}, H_{k+1}]$.

Используемый робастный метод гарантированного эллипсоидального оценивания имеет вид.

$$\begin{aligned} \hat{x}_{k+1} &= \psi(\hat{x}_k, H_k, y_k, p), \\ H_{k+1} &= \Psi(\hat{x}_k, H_k, y_k, p) \end{aligned}$$

где p – вектор параметров алгоритма.

Аппаратная реализация расчета нового эллипсоида

Архитектуру блока для расчета нового эллипсоида (рис.1) можно представить в виде 3-х узлов. Синхронизация работы узлов выполняется за счет единичных сигналов CE (clock enable) и RDY (ready), которые активны на протяжении одного такта и указывают на начало работы и готовность результата. Каждый из узлов содержит мультиплексоры и АЛУ, предназначенные для реализации макрофункций и бинарных операций над данными в формате с плавающей точкой. Выбранная архитектура для блока дает возможность оптимизировать аппаратно-временные характеристики и масштабировать устройство для обработки матриц больших порядков.

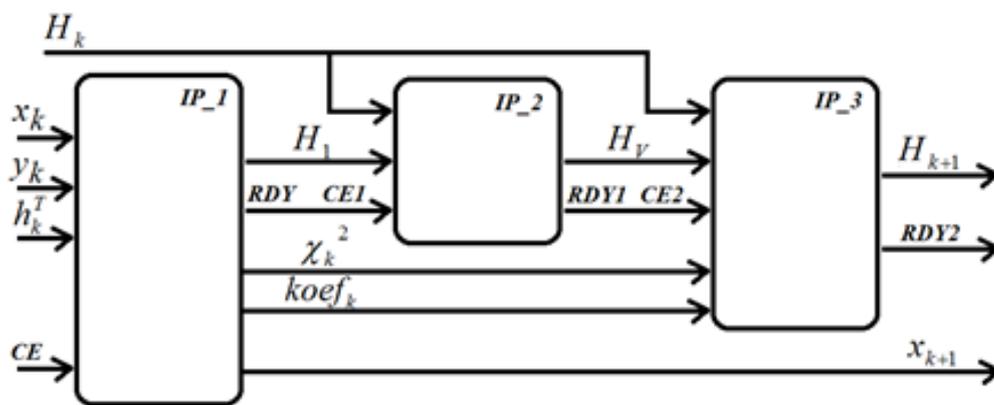


Рис. 1. Архитектура блока для расчета нового эллипсоида

Для упрощения структуры используются макрофункции, реализующие более 1 бинарной операции. В блоке используются такие типы макрофункций:

$$\begin{aligned} &(X_1 \times X_2 - Y_1 \times Y_2); \\ &(X_1 \times X_2 + Y_1 \times Y_2 + Z_1 \times Z_2); \\ &(X_1 \times X_2 + Y_1 \times Y_2 + Z_1 \times Z_2 + M_1 \times M_2). \end{aligned}$$

Данные в макрофункциях обрабатываются потоком (конвейером) за 2 и 3 такта соответственно (рис.2).

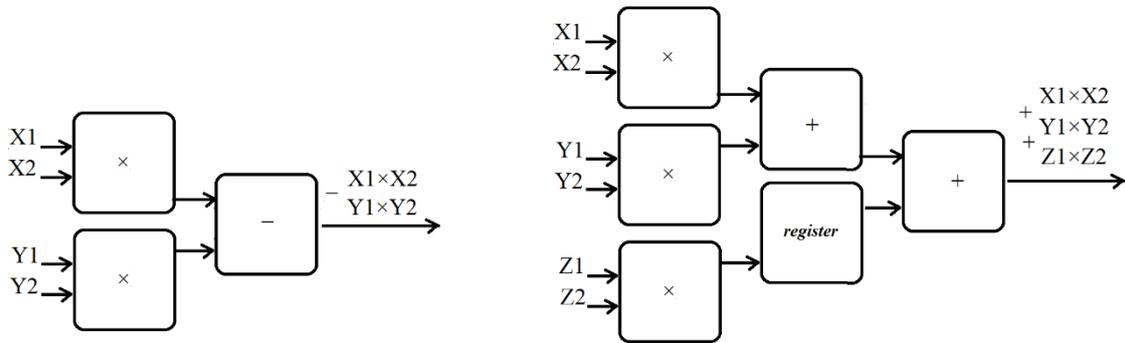


Рис. 2. Структуры макрофункций

Узел IP_1 содержит мультиплексор для поочередной выборки входных данных для их последующей обработки в АЛУ (макрофункция $(X_1 \times X_2 + Y_1 \times Y_2 + Z_1 \times Z_2 + M_1 \times M_2)$) и бинарных операций вычитания, вычисления обратного значения и операций умножения.

Узел IP_1 выполняет вычисления промежуточных коэффициентов $(\chi_k^2, coef_k)$, матрицы H_1 и значения центра эллипсоида на текущем шаге:

χ_k^2 – рассчитывается согласно формуле в постановке задачи,

$$coef_k = \chi_k^2 \times (1 - \beta^2) \times \rho \times s_k^{-2},$$

$$H_1 = H_k \times h \times h^T.$$

Узел IP_2 содержит мультиплексор для поочередной выборки элементов матриц H_k и H_1 для их последующей обработки в АЛУ (макрофункция $(X_1 \times X_2 + Y_1 \times Y_2 + Z_1 \times Z_2 + M_1 \times M_2)$).

Узел IP_2 выполняет промежуточное перемножение матриц $H_V = H_1 \times H_k$.

Узел IP_3 выполняет последний этап вычисления размера нового эллипсоида $H_{k+1} = \chi_k^2 \times H_k - coef \times H_V$. Также узел IP_3 содержит мультиплексор для поочередной выборки элементов.

Тестирование

Тестирования разработанного компонента выполняется путем сравнения и

анализа результатов, полученных после тестирования программной С-модели и тестирования разработанных компонентов в системе моделирования ModelSim [10].

В программной модели используются форматы с плавающей точкой (ФПТ) float (32 bit), double (64 bit). Поскольку обнаружено значительное влияние на накопление погрешности в результате вычисления и интерпретации входных данных в формате Single (32 bit) то компонент реализовано в 2-х вариантах: с поддержкой 32 битной арифметики и 64 битной арифметики.

Исходные данные (x_k, H_k, y_k) . Где x_k – центр эллипсоида на предыдущем шаге, H_k – размер эллипсоида, y_k – вектор измерений:

$$x_k = \begin{pmatrix} x_{1,k} \\ x_{2,k} \\ x_{3,k} \\ x_{4,k} \end{pmatrix}, y_k = \begin{pmatrix} y_{1,k} \\ y_{2,k} \\ y_{3,k} \end{pmatrix},$$

$$H_k = \begin{pmatrix} H_{k_11} & H_{k_12} & H_{k_13} & H_{k_14} \\ H_{k_21} & H_{k_22} & H_{k_23} & H_{k_24} \\ H_{k_31} & H_{k_32} & H_{k_33} & H_{k_34} \\ H_{k_41} & H_{k_42} & H_{k_43} & H_{k_44} \end{pmatrix}.$$

Для расчетов используется входная величина:

$h_k^T = (h_{k_11} \ h_{k_12} \ h_{k_13} \ h_{k_14})$ – для одного измерения;

$$h_k^T = \begin{pmatrix} h_{k_11} & h_{k_12} & h_{k_13} & h_{k_14} \\ h_{k_21} & h_{k_22} & h_{k_23} & h_{k_24} \end{pmatrix} \text{ – для двух}$$

измерений;

$$h_k^T = \begin{pmatrix} h_{k_11} & h_{k_12} & h_{k_13} & h_{k_14} \\ h_{k_21} & h_{k_22} & h_{k_23} & h_{k_24} \\ h_{k_31} & h_{k_32} & h_{k_33} & h_{k_34} \end{pmatrix} \text{ – для трех}$$

измерений.

Промежуточные данные рассчитываются согласно формул: $s_k^2 = h^T H_k h$, $\tilde{y}_k = y_k - h^T x$, а центр нового эллипсоида по формуле: $x_{k+1} = x_k + \rho H_k h s_k^{-2} \tilde{y}_k$, где $\rho \in (0,1)$ – число, параметр оценивания алгоритма. Для расчета матрицы H_{k+1} используем промежуточные коэффициенты:

$$\mu_k = \tilde{y}_k^T s_k^{-2} \tilde{y}_k, \alpha_k = \begin{cases} 1, & \text{если } \mu_k \leq \delta, \\ 1 + \left(1 + \frac{\bar{\rho}}{1 + \det(H_k)}\right) \mu_k & \text{иначе} \end{cases}, \chi_k^2 = \alpha_k - \rho \mu_k;$$

где $\rho \in (0,1)$, $\delta \in (0,1)$, $\beta \in (0,1)$ – параметры алгоритма.

Размер нового эллипсоида рассчитывается по формуле:

$$H_{k+1} = \chi_k^2 [H_k - (1 - \beta^2) \rho H_k h s_k^{-2} h^T H_k].$$

Результаты тестирования

В формулах используются следующие параметры алгоритма оценивания: $\bar{\rho} = 1.5$, $\rho = 0.25$, $\delta = 0.2$, $\beta = 0.95$, и входные данные.

центр эллипсоида x_k	вектор измерений y_k	входная величина h_k^T			
0.02828142201374038	-14946908.607228	0	-141586504.833	-141586504.833	0
0.03617255621237455	-14946908.607228	-141586504.833	0	-141586504.833	0
0.04795394617158478	-14946908.607228	-245518502.24	-245518502.24	0	0
0.00572804184953147					

размер эллипсоида H_k			
2.538488342260649700	-0.026273491941987220	-0.2505030277855324600	1.244924829759154900
-0.02627349194198722	2.1078169540535301000	-0.0066989601734539217	-0.01029679112739984
-0.25050302778553252	-0.006698960173453920	2.2503801682486579000	-0.08974502131259445
1.244924829759154900	-0.010296791127399847	-0.0897450213125944720	0.977102118138873820

Результаты, полученные программной моделью для формата Double (64 bit) совпали с результатами аппаратной реализации.

центр нового эллипсоида x_{k+1}
0.05134680954554588000
0.06452317788614574100
0.05791536557262300100
0.01734862682656122800

Результаты, полученные программной моделью, используя формат Double, можно использовать для анализа точности вычислений 32-х битной реализации аппаратной реализации. Погрешность в аппаратной реализации с поддержкой 32 битного ФПТ имеет две составляющие:

погрешность интерпретации числа из формата Double в формат Single и накопленную в процессе вычисления на 32 битном ФПТ вместо 64 битного. Можно сделать вывод, что интерпретация числа из формата Double в формат Single вносит довольно существенную погрешность в конечный результат.

Незначительное отличие результатов, полученных на программной модели и аппаратной модели с поддержкой ФПТ 64-бит может быть обусловлено различной последовательностью выполнения операций. В случае использования одинаковой последовательности выполнения

операций результаты должны полностью совпадать.

размер нового эллипсоида H_{k+1}			
1.92347051762941930	-0.81474028985023517	-1.0505580121710645	0.929471455302509190
-0.81474028985023494	1.46180173011029630	-0.70519044764678829	-0.409374397337682990
-1.050558012171064	-0.70519044764678818	1.70604562012936260	-0.492296851236767530
0.929471455302509190	-0.40937439733768299	-0.49229685123676759	0.815353084599115820

Выводы

Разработано экспериментальное ядро современного проблемно-ориентированного процессора в элементном базисе FPGA для аппаратной реализации робастных алгоритмов оценивания и управления ДС, которые соответствуют современным тенденциям применения и развития прецизионных систем управления МКА.

Аппаратный компонент работает на частоте 50–60 МГц в FPGA семейства Virtex4 и 90–100 МГц в ПЛИС семейства Virtex7. Для вычисления алгоритма необходимо 100 тактов (общее время выполнения алгоритма около 2 мкс). Погрешность, которая присутствует в результатах вычислений компонентом, есть следствие использования 32 битных вычислений. В случае использования обработки данных с 64-битной арифметикой, погрешность между аппаратной и программной моделями будет отсутствовать.

Выбор формата 32 бита или 64 бита для обработки данных следует делать в зависимости от требований к погрешности вычислений. В случае использования 64 битной арифметики время выполнения алгоритма увеличится примерно в 2 раза, также увеличится характеристика аппаратных ресурсов для компонента.

При необходимости уменьшения времени выполнения алгоритма архитектура компонента может быть расширена за счет добавления дополнительных математических функций. Так как задача хорошо распараллеливается, то время выполнения может быть сокращено до 10 раз (при максимальном распараллеливании).

В частном случае конечную архитектуру компонента следует выбирать, вводя ограничения на одну или несколько характеристик, таких как: аппаратные затраты,

время выполнения, желаемую погрешность, также следует учитывать протокол интерфейса ввода/вывода.

Литература

1. *Kuntsevich V.M., Volosov V.V.* Ellipsoidal and estimation of state vectors for the families of linear and nonlinear discrete-time dynamic systems. *Cybernetics and Systems Analysis*. Springer New York Publishers, 2015. – Vol. 51. – № 1. – P. 64-73.

2. *Schmidt S.F.* The Kalman Filter: Its Recognition and Development for Aerospace Applications. *Journal of Guidance and Control*, 2015. – Vol. 4. – № 1. – P. 4-7.

3. *Palagin A.V., Opanasenko V.N.* Reconfigurable Computing Systems. – К.: Prosvita, 2006. – 295 p.

4. *Опанасенко В.Н.* Высокопроизводительные реконфигурируемые компьютеры на базе FPGA. Проблемы інформатизації та управління, 2009. – Т. 3, вып. 27. – С. 114-118.

5. *Palagin A., Opanasenko V.* The implementation of extended arithmetic's on FPGA-based structures. *Proceedings of the 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*. – Vol. 2. – IDAACS'2017, 21-23 September 2017, Bucharest, Romania. – P. 1014-1019.

6. *Opanasenko V.N., Kryvyi S.L.* Synthesis of Neural-Like Networks on the Basis of Conversion of Cyclic Hamming Codes. *Cybernetics and Systems Analysis*, 2017. – Vol. 53. – № 4. – P. 627-635.

7. *Andrew Good.* "Mars Buggy" Curiosity Measures a Mountain's Gravity", February 5, 2019. Available at <http://mars.nasa.gov/news/8406/mars-buggy-curiosity-measures-a-mountains-gravity/>.

8. Andrew Good, JoAnna Wendel. "Beyond Mars, the Mini MarCO Space-craft Fall Silent", February 5, 2019.

9. Tobias Lange, Holger Michel, Björn Fieth, Harald Michalik. Solar Orbiter Will Process Data Onboard Using Xilinx FPGAs.

Xcell Journal, Xilinx Inc, 2015. – Vol. 90. – № 1. – P. 18-21.

10. ModelSim tutorial. Software Version 10.1c. Mentor Graphics Corporation. 1991-2012. – 81 p. /Available at http://www.microsemi.com/document-portal/doc_view/131618-modelsim-tutorial.

Опанасенко В.Н., Завьялов С.Б., Софіюк О.Т.

АППАРАТНОЕ ЯДРО НА БАЗЕ FPGA ДЛЯ РОБАСТНЫХ АЛГОРИТМОВ ОЦЕНКИ ВЕКТОРА СОСТОЯНИЯ И УПРАВЛЕНИЯ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

Рассмотрена аппаратная реализация ядра на базе FPGA для робастных алгоритмов оценки вектора состояния и управления динамических систем. Проведено сравнение и анализ результатов расчета нового центра эллипсоида, полученных для модели C – программы, а также разработанного аппаратного ядра для оценки эллипсоидального состояния и алгоритмов управления. Разработка осуществлена с использованием системы проектирования WebPack ISE. Тестирования разработанного ядра выполняется путем сравнения и анализа результатов, полученных после тестирования программной C-модели и тестирования разработанного ядра в системе моделирования ModelSim. В программной модели используются форматы с плавающей точкой (ФПТ) Single (32 bit), Double (64 bit). Поскольку обнаружено значительное влияние на накопление погрешности в результате вычисления и интерпретации входных данных в формате Single (32 bit), то аппаратное ядро реализовано в 2-х вариантах: с поддержкой 32 битной арифметики и 64 битной арифметики.

Ключевые слова: управление динамическими системами, оценки, FPGA, малый космический аппарат.

Opanasenko V.M., Zavyalov S.B., Sofiuk O.T.

FPGA-BASED HARDWARE CORE FOR ROBUST ALGORITHMS FOR ASSESSING THE STATE VECTOR AND CONTROL OF DYNAMIC SYSTEMS

The hardware FPGA-based implementation of the Core for robust algorithms for estimating the state vector and controlling dynamic systems is considered. Comparison and analysis of the results of calculating the new center of the ellipsoid obtained for the model C - program, as well as the developed hardware kernel for evaluating the ellipsoidal state and control algorithms are carried out. The development was carried out using the WebPack ISE design system. Testing of the developed core is performed by comparing and analyzing the results obtained after testing the software C-model and testing the developed core by means of the ModelSim modeling system. The software model uses floating point formats – Single (32 bit), Double (64 bit). Since a significant influence on the accumulation of errors was found as a result of calculating and interpreting input data in the Single (32 bit) format, the hardware core is implemented in 2 versions: with support for 32 bit arithmetic and 64 bit arithmetic.

Key words: control of dynamic systems, estimation, FPGA, on-board processor, small spacecraft.