ACTA IMEKO
ISSN: 2221-870X
December 2019, Volume 8, Number 4, 28 - 32

Hierarchical data fusion architecture for autonomous systems

Ivan Ermolov[1]

[1] *Ishlinsky Institute for Problems in Mechanics of the Russian Academy of Sciences, Russia*

ABSTRACT
Autonomy is becoming a key issue concerning unmanned vehicles nowadays. An effective functioning of an unmanned system requests the processing of a large amount of data coming from sensors, onboard databases, etc. Therefore, data fusion technology is a key technology used in autonomous systems. In order to systemise such data processing in autonomous systems, special so-called data fusion architectures are used (e.g. JDL, Waterfall, Boyd). However, some of those solutions have many restrictions. A goal of this study is to present a novel hierarchical data fusion architecture that can be used in autonomous systems. This architecture consists of five basic layers: identification of parameters, state identification, object type identification, situation identification, and task implementation identification. The proposed architecture has some advantages in comparison to those that are already in use. The author considers that the presented architecture has good visibility; intuitive understanding; the possibility of deep feedback usage; and good potential for automatic reconfiguration and self-learning. The developed data fusion architecture can be used for building complex data fusion systems on board of autonomous systems, of a group of unmanned vehicles, and even of systems of a higher hierarchy.

Section: RESEARCH PAPER

Keywords: data fusion; sensor fusion; unmanned vehicles; autonomous systems

Citation: Ivan Leonidovich Ermolov, Hierarchical data fusion architecture for autonomous systems, Acta IMEKO, vol. 8, no. 4, article 6, December 2019, identifier: IMEKO-ACTA-08 (2019)-04-06

Editor: Yvan Baudoin, International CBRNE Institute, Belgium

Received November 23, 2018; **In final form** September 12, 2019; **Published** December 2019

Funding: This work was supported by Russian Ministry of Science and Higher Education (Project Reg. No. AAAA- A17-117021310384-9)

Corresponding author: Ivan L. Ermolov, e-mail: ermolov@ipmnet.ru

1. INTRODUCTION

Autonomy is a feature that modern robotic systems severely lack. In robotics, autonomy can be understood as the ability of a robot to function for an extended period of time, over broad spaces, and in unpredictable environments without the need to collaborate with friendly objects or human operators.

In [2], the topic of robotic autonomy was discussed. The need for an autonomous system is urgent for the following reasons:

- Autonomous systems, as a rule, are more efficient than human-operated systems,
- The expansion of the application areas of robotic systems,
- The abandonment of involvement of high-skilled human operators,
- The possibility of using robots in groups [11], [12],
- Wide usage of optimisation and complex algorithms,
- The abandonment of redundant components from robotic systems,
- Minimisation of the mass and size of robots,
- Decreasing time delays in robots' functioning, and
- Predictability of robots' behaviour.

It is postulated herein that in order to implement the autonomy of unmanned vehicles, it is efficient to increase their information supply and intelligence level.

Both of these elements are connected with data processing coming from sensors of the robot and the environment. Due to the specifics of the sensors used in robots, it is necessary to use data fusion. The importance of data fusion is emerging for mobile systems in particular.

2. DATA FUSION IN ROBOTICS

Data fusion in itself has been widely known for centuries. Humans widely use data fusion (fusion of the five senses, distribution and doubling of sensors, etc.).

Much interesting research has been undertaken on data fusion, especially in robotics [9], [13].

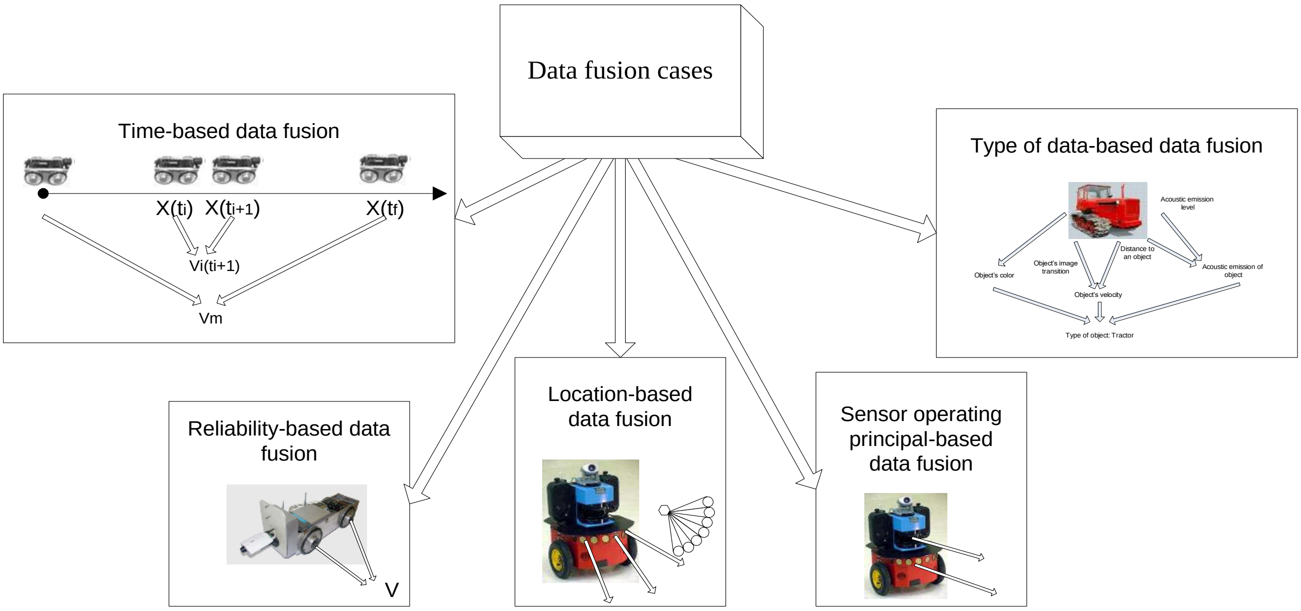
Data fusion cases
Time-based data fusion
X(ti)
X(ti+1)
X(tf)
Vi(ti+1)
Vm
Type of data-based data fusion
Acoustic emission level
Object's color
Object's image transition
Distance to an object
Acoustic emission of object
Object's velocity
Type of object: Tractor
Reliability-based data fusion
V
Location-based data fusion
Sensor operating principal-based data fusion

Figure 1. Basic cases of data fusion in robotics.

Data fusion [14], [15], [17] in itself can be defined as a process of information generalisation based on more than one source of information.

It is important to note the following advantages of data fusion for robotists:

- It is cheaper to produce new information by developing software than by installing extra sensors,
- Considerable energy savings because information processing requires less energy than the sensor hardware,
- Minimisation of the autonomous systems' mass and sizes, which is especially crucial for autonomous systems working in severe environments [3],
- Decrease of wires and other interfaces, and thus, higher reliability,
- Decrease of negative interference among autonomous systems' components,
- The possibility of using lower performance (i.e. cheaper) sensors with a higher quality of information therefrom,
- Compensation for the restricted working space and spectrum of sensors,
- Transfer of the computational load from the human operator to the onboard control system,
- Increased unification by the use of the same set of sensors for various functional tasks.

The author's proposal is outlined in Figure 1 to distinguish the following cases of data fusion in robotics:

Time-based data fusion. While tracking the variation of some parameters in time sequences, it is possible to estimate other parameters of the autonomous system. Another case of such data fusion is data filtering. For more details see [4].

Reliability-based data fusion. While fusing data from several sensors with low-reliability characteristics, it is possible to get highly reliable information.

Space-based data fusion. By fusing information from several sensors, each one with a narrow working space (example in Figure 2), we may get information on the large working space. Another case of space-based data fusion is fusing data from dispensed sensors, which gives us new information.

Sensor-type data fusion. Fusion of data from sensors measuring the same parameter but functioning on various principles gives information with a higher reliability factor (Figure 3).

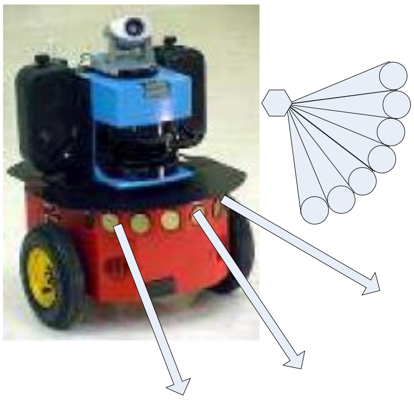

Figure 2. Space-based data fusion (fusion of ultrasonic data using the Pioneer P3DX platform by Adept Mobilerobots).

Figure 3. Navigation data from various types of sensors is fused directly onboard the 120/3 navigation system from Perm Instruments Making Company.

Data-type data fusion. This type is used to produce information based on data of various types (e.g. fusing information from a video sensor with information from a laser sensor). Such fusion is intensively used, especially for object recognition [7].

In sum, most of the data fusion cases in unmanned systems can be described by these fusion cases or by a combination thereof.

3. HIERARCHICAL DATA FUSION ARCHITECTURE

Modern autonomous systems are extremely complex, with a large number of data flows [5]. In order to analyse the large number of data flows, it is necessary to adjust those data flows. This is known as data fusion architectures. There is a variety of such architectures: JDL, Boyd model, LAAS, The Omnibus Model, the Waterfall model, and more [4], [6], [16]. By the way, these architectures are also used for other applications (see e.g. [18], [19]).

However, according to the author, these architectures lack some intuitiveness, logic, and flexibility.

The JDL architecture has some restrictions [6]:

- Most of the models were organised based on some specific data or information, e.g. it is difficult to combine a JDL model for other applications,
- The outlook of the model seems to be rather abstract, which creates an obstacle to its interpretation,
- This architecture does not correlate with some specific data processing algorithms, which impedes its implementation in real systems.

Concerning data fusion architecture that is mostly suitable for unmanned vehicles, one may formulate the following criteria for such architecture:

- Data fusion architecture should have a hierarchical structure, as only a hierarchy allows for the control of large-scale systems,
- Data fusion architecture should demonstrate all processes in their hierarchy, be easily understandable, and even be intuitive for its user,
- Such architecture should allow various feedback and counter-current data flows,
- In some cases, the architecture should permit data transfer, omitting some hierarchical layers.

The author proposes the hierarchical data fusion architecture shown in Figure 4.

The layers' terms are as follows [8]:

Parameter – the predicate that describes, as a rule, the quantitative property of a single component of an autonomous system or of the environment. The input for this level usually comes directly from the sensors.

Mathematically, the parameter can be described as follows:

$$p_i = \sum_{j=1}^{z} a_{ji}^{gp} g_j + \sum_{k=1}^{y} a_{ki}^{pp} p_k \,, \tag{1}$$

where p_i – *i*-th parameter; a_{ji}^{gp} – the coefficient considering the influence of the *j*-th sensor on the *i*-th parameter; g_j – input from the *j*-th sensor; *z* – number of sensors in the system; a_{ki}^{pp} – the coefficient considering the influence of the *k*-th parameter on the *i*-th parameter (in *k=i* it is predicate's 'inertia'); p_k – the value of the *k*-th parameter; and *y* – the number of parameters in system.

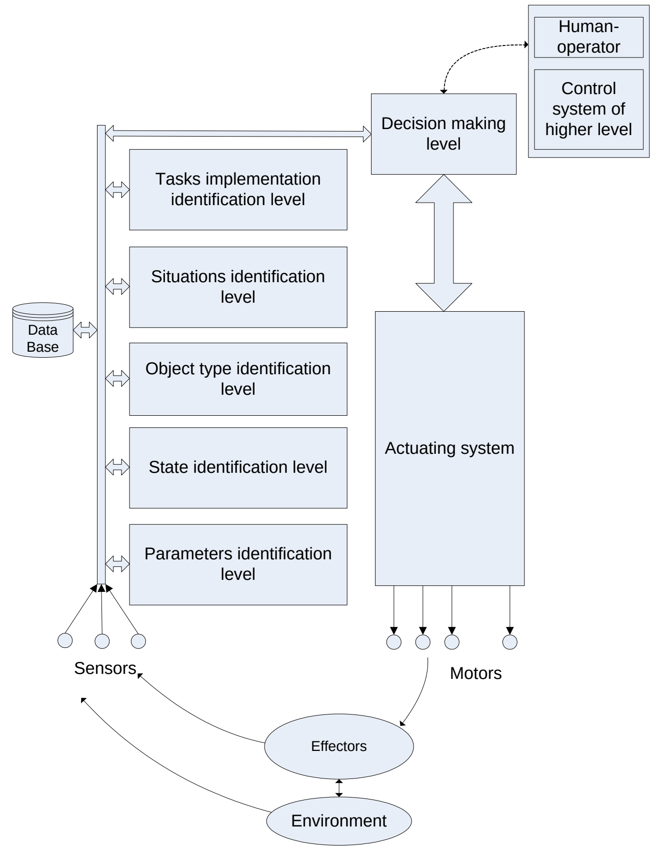
Human-operator
Control system of higher level
Decision making level
Tasks implementation identification level
Situations identification level
Data Base
Object type identification level
State identification level
Parameters identification level
Actuating system
Sensors
Motors
Effectors
Environment

Figure 4. Hierarchical data fusion architecture.

State – the predicate that describes quantitatively or relatively the property of the whole autonomous system or of the environment. The input for this predicate comes directly from the parameters or from the sensors.

Mathematically, it can be described as follows:

$$s_i = \sum_{j=1}^{y} a_{ji}^{ps} p_j + \sum_{k=1}^{x} a_{ki}^{ss} s_k + \sum_{h=1}^{w} a_{hi}^{os} o_h \,, \tag{2}$$

where s_i – *i*-th state; a_{ji}^{ps} – the coefficient considering the influence of the *j*-th parameter on the *i*-th state; a_{ki}^{ss} – the coefficient considering the influence of the *k*-th state on the *i*-th state; o_h – *h*-th instruction; *x* – the number of states in the system; a_{hi}^{os} – the coefficient considering the influence of the *h*-th instruction on the *i*-th state; and *w* – the number of instructions in system.

Object type – the generalised identification of an object present in the environment, defined by its typical data and by its potential interaction with the autonomous system.

Mathematically, it can be described as follows:

$$m_i = \sum_{j=1}^{y} a_{ji}^{pm} p_j + \sum_{k=1}^{x} a_{ki}^{sm} s_k + \sum_{h=1}^{v} a_{hi}^{mm} m_h + \sum_{l=1}^{u} a_{li}^{cm} c_l + \\ + \xi_i^m + + \psi_i^m \,, \tag{3}$$

where m_i – the *i*-th type of object; a_{ji}^{pm} – the coefficient considering the influence of the *j*-th parameter on the identification of the *i*-th type of objects; a_{ki}^{sm} – the coefficient considering the influence of *k*-th state on the identification of the *i*-th type of objects; a_{hi}^{mm} – the coefficient considering the influence of various other types of objects presented in the environment to the identification of the *i*-th object type; *v* – the number of object type predicates in the system; a_{li}^{cm} – the coefficient considering the influence of the *l*-th state identification of the *i*-th object type; c_l – the *l*-th situation predicate; *u* – the number of situations in the system; ξ_i^m – information from the database on the trends of the *i*-th type of objects' presence; ψ_i^m – information from tasks on the possibility of the *i*-th type of objects' presence.

Situation – a generalised notion that describes the combination of interactions between the robot and the environment.

Mathematically, it can be described as follows:

$$c_i = \sum_{j=1}^{y} a_{ji}^{pc} p_j + \sum_{k=1}^{x} a_{ki}^{sc} s_k + \sum_{h=1}^{v} a_{hi}^{mc} m_h + \sum_{l=1}^{u} a_{li}^{cc} c_l + \\ + \xi_i^c + \psi_i^c, \qquad (4)$$

where c_i – the *i*-th situation, a_{ji}^{pc} – the coefficient considering the influence of the *j*-th parameter on the identification of the *i*-th situation; a_{ki}^{sc} – the coefficient considering the influence of *k*-th state on the identification of the *i*-th situation; a_{hi}^{mc} – the coefficient considering the influence of various types of objects presented on the identification of the *i*-th situation; a_{li}^{cc} – the coefficient considering the influence of the *l*-th situation on the identification of the *i*-th situation; ξ_i^c – information from the database on the trends of the *i*-th situation; and ψ_i^c – information from tasks on the possibility of the *i*-th situation.

Tasks – a set of situations in serial-parallel order. These situations must be achieved (implemented) by the robot in order for a task to be fulfilled.

Mathematically, it can be described as follows:

$$t_i = \sum_{j=1}^{y} a_{ji}^{pt} p_j + \sum_{k=1}^{x} a_{ki}^{st} s_k + \sum_{h=1}^{u} a_{hi}^{ct} c_h + \sum_{l=1}^{q} a_{li}^{tt} t_l + \xi_i^t , \qquad (5)$$

where t_i – the *i*-th task implementation; a_{ji}^{pt} – the coefficient considering the influence of the *j*-th parameter on the identification of the *i*-th task implementation; a_{ki}^{st} – the coefficient considering the influence of the *k*-th state on the identification of the *i*-th task implementation; a_{hi}^{ct} – the coefficient considering the influence of the *h*-th situation on the identification of the *i*-th task implementation; a_{li}^{tt} – the coefficient considering the influence of the *l*-th task implementation on the identification of the *i*-th task implementation; *q* – the number of tasks in the system; and ξ_i^t – information from the database on the trends of the *i*-th task implementation.

The operation of the scheme is now explained.

Usage of a common data bus is also an important advantage as it allows for the free exchange of data and information among various levels. As explained above, data may flow to the next level and to higher levels directly. This also allows for the human operator to receive information on any predicate directly in a case of need.

All the data is fused in a bottom-up direction. However, it is obvious that some simpler or more primitive systems may omit some of the higher levels. However, highly autonomous systems will involve all the levels. When needed, extra data is acquired from the database. The final output comes to the decision-making level, which sends instructions to actuators and sub-systems. It also may be built using the same approach.

The proposed hierarchical data fusion architecture has the following advantages compared to other architectures:

- It reflects the hierarchy of the modern autonomous systems' structure,
- It has clear demonstrable and visual properties, with a high degree of intuition for the human operator,
- It allows the wide usage of various data flows and feedback, including transient flows,
- This structure can be effectively implemented with modern control methods [1],
- The structure's modularity allows various ready-to-use solutions to be transferred from one scheme to another.

It is the author's opinion that the proposed structure can also be widely used for the automated programming of complex data fusion systems and algorithms.

4. MATHEMATICAL IMPLEMENTATION

The question of which mathematical methods are the most appropriate for data fusion has been discussed by many researchers, e.g. [9], [10].

However, as discussed in [20], data fusion deals with various information types and is used for different tasks. Hence, different types of data fusion can be supported by different mathematical methods.

In [20], a special classification was presented, stating the following:

Low-level data fusion is better supported by Kalman filter, figure of merit, and gating techniques.

Mid-level data fusion is more efficiently implemented by Bayesian decision theory, Dempster-Schafer evidential reasoning, and neural networks.

High-level data fusion solutions are realised by expert systems and fuzzy logic.

However, according to the author of this article, it would be better to find some universal method for various levels of data and sensor fusion.

This can be done using the fuzzy cognitive maps method [21].

In fact, the fuzzy cognitive maps method has the following advantages:

- It has good demonstrable and visual properties, with a high degree of intuition for software engineers,
- The structure allows for the wide usage of various data flows and feedback, including transient flows,
- It can deal with noisy signals i.e. it has good filtering capacity,
- The structure's modularity allows various ready-to-use solutions to be transferred from one scheme to another.

Therefore, these properties almost coincide with the properties of the proposed hierarchical data fusion architecture presented in this paper.

In [8], an example is given supporting this statement.

5. CONCLUSIONS

Data fusion is an important technology implementing the high autonomy of an autonomous system. Hierarchical data fusion architecture has good properties of modularity, visualisation, and hierarchy. It allows for the systematisation of data processing in the complex system of modern autonomous systems.

ACKNOWLEDGEMENT

This work was supported by the Russian Ministry of Science and Higher Education (Project Reg. No. AAAA-A17-117021310384-9).

REFERENCES

[1] I. M. Makarov, V. M. Lokhin, Intelligent Control Methods, Physmatlit, Moscow, 2001.

[2] I. L. Ermolov, Robots' autonomy: its measures and how to increase it, Mechantronics, Automation, Control 8 (2008).

[3] A. A. Fomichev, V. B. Uspensky, K. Y. Stchastlivets, Y. Y. Broslavets, A. B. Koltchev, 'Data fusion in integrated navigation system with laser gyroscopes based on generalized Kalman filter', Proc. of the International Conference on Integrated Navigation Systems, 26-28 May 2003, St. Petersburg, Russia.

[4] V. I. Gorodetsky, O. V. Karsaev, V. V. Samoylov, 'Data fusion in complex situations analysis and understanding', Proc. of the Perspective Systems and Control Tasks Conference, 2008, Dombai, Russia.

[5] I. L. Ermolov, Factors affecting UGVs spacious autonomy level, Vestnik YuFU 1 (2016).

[6] W. Elmenreich, Sensor Fusion in Time-Triggered Systems, Technischen Universitat Wien, Wien, im Oktober 2002.

[7] S. L. Zenkevich, A. A. Minin, Mapping by autonomous system with a Lidar basing recurrent filtration method. Mechantronics, Automation, Control 8 (2007).

[8] I. L. Ermolov, Hierarchical Data Fusion Architecture for Unmanned Vehicles, in Smart Electromechanical Systems: The Central Nervous System, Springer, 2017.

[9] D. L. Hall, Mathematical Techniques in Multisensor Data Fusion, Artech House, 2004.

[10] I. R. Goodman, R. P. Mahler, H. T. Nguyen, Mathematics of Data Fusion, Netherlands, 2000.

[11] V. Dashevskiy, V. Budkov, A. Ronzhin, 'Survey of modular robots and developed embedded devices for constructive and computing components', Proc. of the International Conference on Interactive Collaborative Robotics, 2017, Springer, Cham. LNAI 10459.

[12] S. Manko, V. Lokhin, M. Romanov, 'From autonomous robots to multiagent robotic systems', Proc. of ICMT 2012 – the 16th International Conference on Mechatronics Technology, 2012, Tianjin, China, pp. 594-596

[13] W. Elmenreich 'A review on system architectures for sensor fusion applications', in: Software Technologies for Embedded and Ubiquitous Systems (Lecture Notes in Computer Science vol 4761). R. Obermaisser, Y. Nah, P. Puschner, F. J. Rammig (editors). Springer, Berlin, Heidelberg, 2007.

[14] E. P. Blasch, E. Bossé, D. A. Lambert, High-Level Information Fusion Management and System Design, Artech House Publishers, Norwood, MA, 2012.

[15] M. E. Liggins, J. Llinas. D. L. Hall, Handbook of Multisensor Data Fusion: Theory and Practice, CRC Press, 2009.

[16] F. Castanedo, A review of data fusion techniques, The Scientific World Journal 2013, Article ID 704504.

[17] H. B. Mitchell, Multi-sensor Data Fusion – An Introduction, Springer-Verlag, Berlin, Germany, 2007.

[18] D. H. de la Iglesia, G. Villarrubia, J. F. de Paz, J. Bajo, Multi-sensor information fusion for optimizing electric bicycle routes using a swarm intelligence algorithm, Sensors 17(11) (2017) 2501. doi:10.3390/s17112501

[19] M. M. Almasri, K. M. Elleithy, 'Data fusion models in WSNs: comparison and analysis', Proc. of the 2014 Zone 1 Conference of the American Society for Engineering Education.

[20] D. J. Dailey, P. Harn, P.-J. Lin, ITS Data Fusion: Final Research Report, Research Project T9903, Washington State Department of Transportation, April 1996.

[21] C. D. Stylios, P. P. Groumpos, Modeling complex systems using fuzzy cognitive maps, IEEE Transactions on Systems, Man and Cybernetics, January 2004.