

# Hierarchical Data Fusion Architecture for Unmanned Vehicles

I.L. Ermolov

**Abstract** *Purpose* effective functioning of unmanned vehicles demands to process large amounts of various data. In order to systemize such data processing special so-called data fusion architectures are used (e.g. JDL, Waterfall, Boyd etc.). However, those have some shortages restricting their wide usage. A goal of this paper is to develop a new data fusion architecture which could be used on board of unmanned vehicles. *Result* this paper presents new hierarchical data fusion architecture for unmanned vehicles. This architecture has some advantages in comparison to those already in use. *Practical results* the developed data fusion architecture can be used for building complex data fusion systems on board of unmanned vehicles as well as of group of vehicles and even of systems of higher hierarchy.

**Keywords** Data fusion · Sensor fusion · Unmanned vehicles

## 1 Introduction

In [1] a topic of Unmanned Vehicles (UV) autonomy was discussed. It proves emergency to provide UV with autonomy, in first term through multi-agent usage and further expansion of their applications. It is postulated that in order to implement autonomy of Unmanned Vehicles it is efficient to increase their information supply and intelligence level.

Both of these directions are connected with processing of data coming from sensors of UV and of environment. Due to specifics of sensors used in UVs it is necessary to use data fusion. Importance of data fusion is especially emerging for mobile systems.

---

I.L. Ermolov (✉)

Institute for Problems in Mechanics of the Russian Academy of Sciences,  
Moscow State Technological University “STANKIN”, Moscow, Russia  
e-mail: ermolov@ipmnet.ru

© Springer International Publishing AG 2017

A.E. Gorodetskiy and V.G. Kurbanov (eds.), *Smart Electromechanical Systems: The Central Nervous System*, Studies in Systems, Decision and Control 95,  
DOI 10.1007/978-3-319-53327-8\_6

## 2 Data Fusion

Data fusion by itself is widely known for centuries. Actually the most perfect creature, a human widely uses data fusions (fusion of five senses, distribution and doubling of sensors etc.).

Data fusion by itself can be defined as a process of information generalization based on more than once source of information.

Speaking about advantages which data fusion gives to robotists one should mention following:

- it's cheaper to produce new information by developing software than by installing extra sensors;
- considerable energy savings as information processing requires less energy than hardware of sensors;
- minimization of UVs' mass and sizes, this is especially crucial for UVs working in severe environment;
- decrease of wires and other interfaces, this higher reliability;
- decrease of negative interference among UVs' components;
- possibility to use lower performance (i.e. cheaper) sensors with higher quality of information got from them;
- compensation of restricted working space and spectrum of sensors;
- transfer of computational load from human-operator to on-board control system;
- unification increase through using same set of sensors for various functional tasks.

## 3 Data Fusion in UVs

In this paper it's proposed to distinguish following cases of data fusion in Unmanned systems:

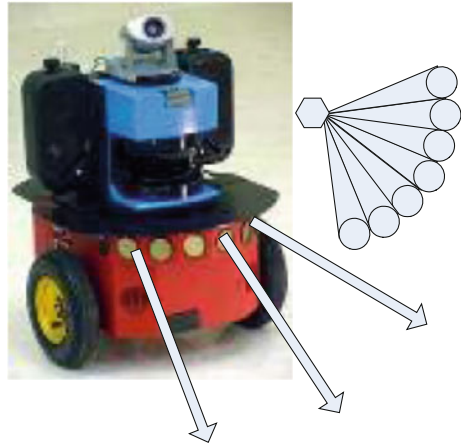
*Time-based data fusion.* While tracking variation of some parameter in time sequences it becomes possible to estimate other parameters of UV. Another case of such data fusion is data filtering [2]. For more details see [3].

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

*Space-based data fusion.* By fusing information from several sensors each one with narrow working space (Fig. 1) we may get information of 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 same parameter but functioning on various principles gives information with higher reliability factor (Fig. 2).

**Fig. 1** Space-based data fusion (fusion of ultrasonics data at Pioneer 3-DX platform by Adept Mobilerebots)



**Fig. 2** Navigation data from various types of sensors is fused directly on board of 120/3 navigation system from Perm Instrument Making Company



*Data-type data fusion.* This one is used to produce information basing on data of various type (e.g. fusing information from video sensor with information from laser sensor). This type of fusion is intensively used especially for object recognition.

Summarizing this classification it can be said that most of the data fusion cases in unmanned systems can be described by these fusion cases or by their combination.

*Data fusion architectures.* Modern UVs are extremely complex systems with large number of data flows [4]. In order to analyze such amount of data flow it is necessary to adjust those data flows.

This is known as data fusion architectures. There exists variety of such architectures: JDL, Boyd model, LAAS, The Omnibus Model, Waterfall model and some others [5].

One of the most popular solutions in this area was the JDL (Joint Directors of Laboratories) Architecture [6] which was developed in 1985.

This architecture includes 5 level data hierarchy and a data base connected with common bus. Those levels may act both in sequence and in parallel.

However JDL architecture has some restrictions [6]:

- each time model is organized basing on some specific data or information. E.g. it's difficult to recombine a JDL-model for other applications.
- Outlook of the model seems to be rather abstract, what creates an obstacle for its interpretation;
- This architecture doesn't correlate with some specific data processing algorithms, this impedes its implementation in real system.

Speaking of data fusion architecture mostly suitable for Unmanned Vehicles one may formulate following criteria for such architecture:

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

## 4 Hierarchical Data Fusion Architecture

Hierarchy is inherent with some data fusion architectures presented above. However the distinction between layers is not demonstrative there.

Author of this paper proposes following *Hierarchical data fusion architecture* (Fig. 3).

Here author describes layers' term as following:

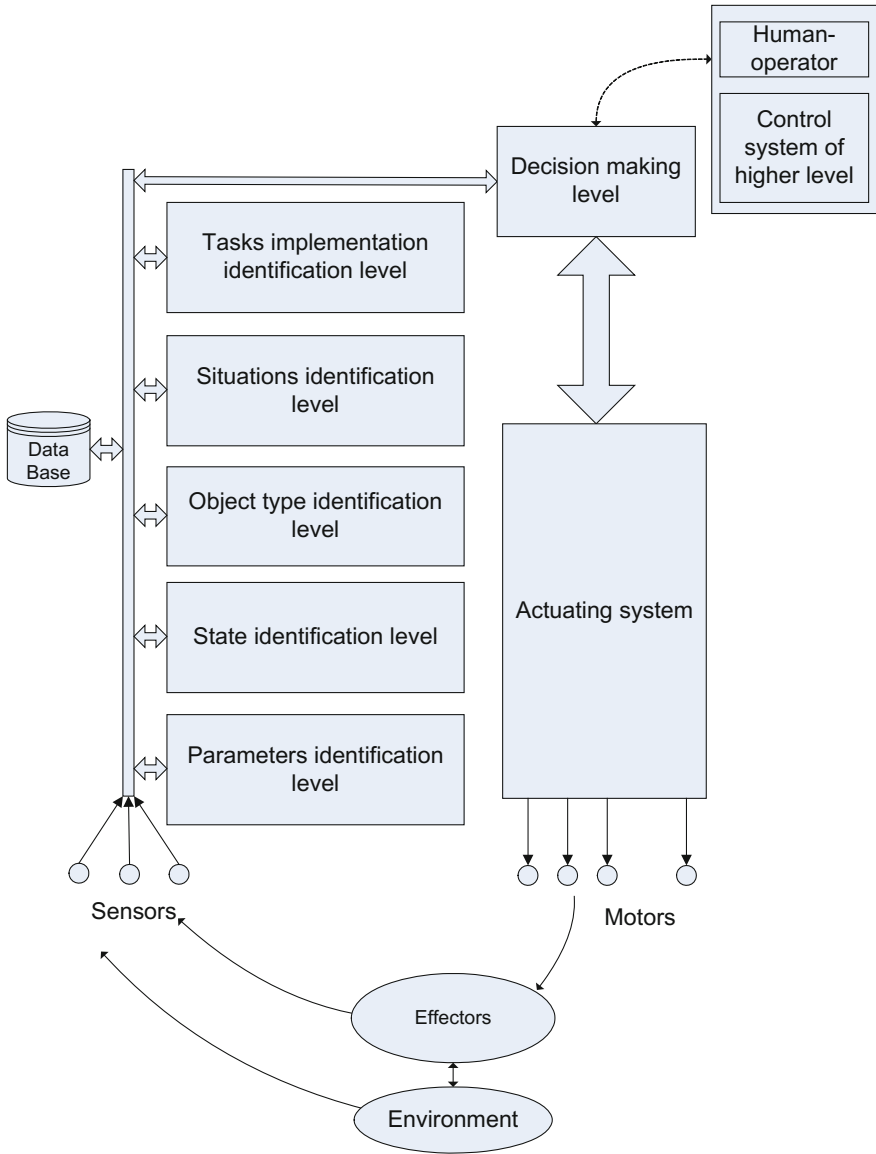
*Parameter*—predicate which describes as a rule quantitatively property of a single component of a UV or of environment. Input of this level usually comes directly from sensors.

Mathematically it can be described as following:

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

where

- $p_i$  i-th parameter,
- $\alpha_{ji}^{gp}$  coefficient considering influence of j-th sensor to i-th parameter,
- $g_j$  input from j-th sensor,
- $z$  number of sensors in system,



**Fig. 3** Hierarchical data fusion architecture

- $a_{ki}^{pp}$  coefficient considering influence of k-th parameter to i-th parameter (in  $k = i$  it is predicate's "inertia"),
- $p_k$  value of k-th parameter,
- $y$  number of parameters in system.

*State*—predicate which describes quantitatively or relatively property of a whole UV or of environment. Input for this predicate comes from *parameters* or from sensors directly.

Mathematically it can be described as following:

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

where

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

*Object type*—generalized identification of an object present in environment, which is defined by its typical data and by its potential interaction with UV.

Mathematically it can be described as following:

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

where

- $m_i$  i-th type of object,
- $a_{ji}^{pm}$  coefficient considering influence of j-th parameter on identification of i-th type of objects,
- $a_{ki}^{sm}$  coefficient considering influence of k-th state on identification of i-th type of objects,
- $a_{hi}^{mm}$  coefficient considering influence various other types of objects presented in environment to the identification of i-th object type,
- $v$  number of objects type predicates in system,
- $a_{li}^{cm}$  coefficient considering influence of l-th state identification of i-th object type,
- $c_l$  l-th situation predicate,
- $u$  number of situations in system,
- $\zeta_i^m$  information from data base on trends of i-th type of objects presence,
- $\psi_i^m$  information from tasks on the possibility of i-th type of objects presence.

*Situation*—is a generalized notion which describes complex of interaction between robot and environment.

Mathematically it can be described as following:

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

where

$c_i$  i-th situation

$a_{ji}^{pc}$  coefficient considering influence of j-th parameter on identification of i-th situation,

$a_{ki}^{sc}$  coefficient considering influence of k-th state on identification of i-th situation,

$a_{hi}^{mc}$  coefficient considering influence of various types of objects presented on identification of i-th situation,

$a_{li}^{cc}$  coefficient considering influence of l-th situation on identification of i-th situation,

$\zeta_i^c$  information from data base on trends of i-th situation,

$\psi_i^c$  information from tasks on the possibility i-th situation.

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

Mathematically it can be described as following:

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

where

$t_i$  i-th task implementation,

$a_{ji}^{pt}$  coefficient considering influence of j-th parameter on identification of i-th task implementation,

$a_{ki}^{st}$  coefficient considering influence of k-th state on identification of i-th task implementation,

$a_{hi}^{ct}$  coefficient considering influence of h-th situation on identification of i-th task implementation,

$a_{li}^{tt}$  coefficient considering influence of l-th task implementation on identification of i-th task implementation,

$q$  number of tasks in system,

$\zeta_i^t$  information from data base on trends i-th task implementation

Now let's see how it works.

Data from various sensors of UV and of environment flows to the common bus. An important advantage of this scheme is that it can use data from both on-board and stand-alone sensors.

Usage of common data bus is also an important advantage as it allows to exchange data and information among various levels freely. As it was mentioned before data may flow to the next level and also to higher levels directly.

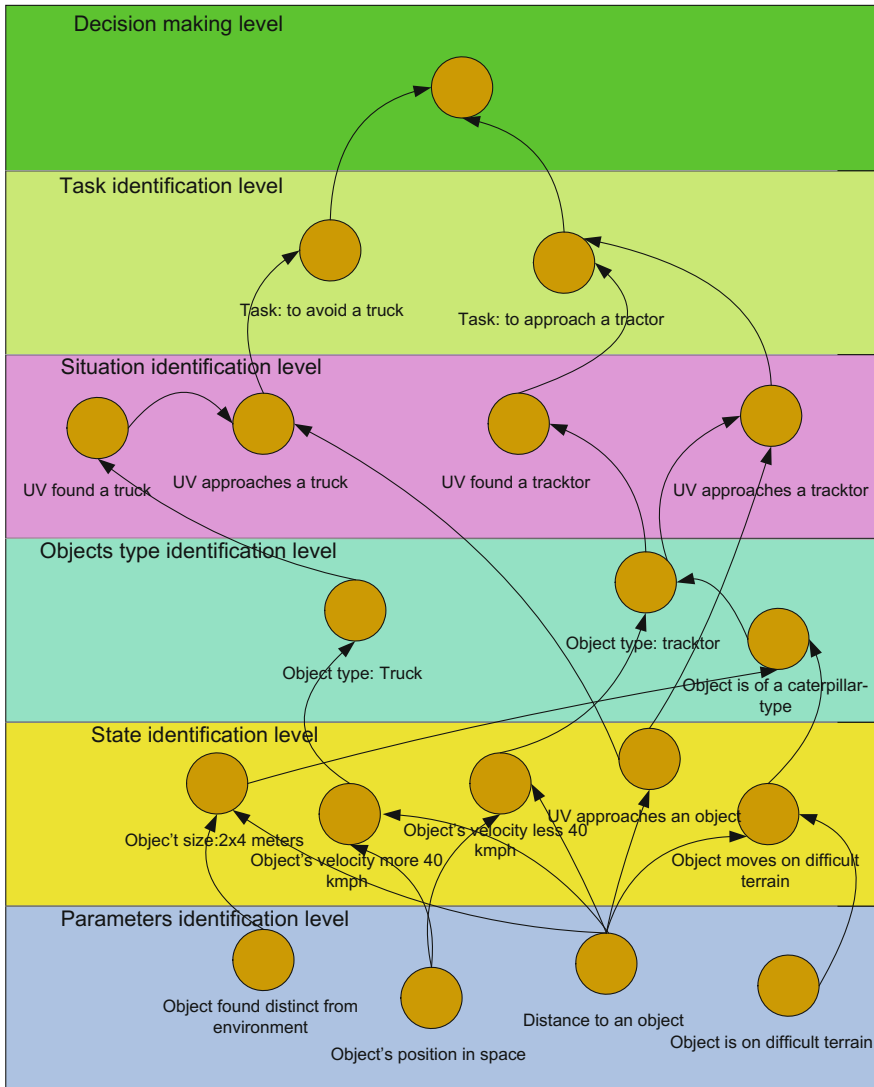


Fig. 4 Example of a model built upon Hierarchical data fusion architecture

This also allows in case human-operator needs it to receive information on any predicate directly.

All the data is fused in bottom-up direction. However it's obvious that some more simple or primitive systems may omit some of the higher levels. However highly autonomous systems will involve all the levels.

When needed extra data will be acquired from data base.

Final output comes to the decision making level which sends instructions to actuators and sub-systems. It also may be built on same approach.



Figure 4 presents example of model built on *Hierarchical data fusion architecture*. In this example UV has a goal to approach tractor, while avoiding a truck. This scheme shows only some of the data fusion components and data flows.

Let's outline advantages of the proposed *Hierarchical data fusion architecture*:

- it is appropriate to modern UV structure;
- it has good demonstrable and visual properties, which allow operator to use it intuitively;
- wide usage of various data flows and feedbacks including transient flows;
- modularity which allows various ready-to-use solutions be transferred from one scheme to another.

Software implementation of proposed architecture involves intelligent control approaches, such as neural networks, fuzzy cognitive maps, expert systems [7, 8].

## 5 Conclusions

Data fusion is an important technology implementing high autonomy of UV. Hierarchical data fusion architecture has good properties of modularity, visualization and hierarchy. It allows to systemize data processing in complex system of modern UVs.

Research supported by Russia Academy of Sciences grant: “Emerging Problems of Robotics”.

## References

1. Ermolov, I.L.: Robots' Autonomy, its Measures and How To Increase it. *Mechatron. Autom. Control*, № 528 (2008)
2. Zenkevich, S.L., Minin, A.A.: Mapping by UV with a lidar basing recurrent filtration method. *Зенкевич С.Л., Минин А.А., Mechatron. Autom. Control*, № 8 (2007)
3. Fomichev, A.A., Uspensky, V.B., Stchastlivets, K.Y., et al.: Data fusion in integrated navigation system with laser gyroscopes based on generalized Kalman filter. In: *Proceedings of International Conference on Integrated Navigation Systems*, St. Petersburg, 26–28 May 2003
4. Ermolov, I.L.: Factors affecting UGVs spacious autonomy level, *Vestnik YuFU*, № 1, 2016
5. Gorodetsky, V.I., Karsaev, O.V., Samoylov, V.V.: Data fusion in complex situations analysis and understanding. In: *Proceedings of “Perspective Systems and Control Tasks”*, Dombai, Russia (2008)
6. Elmenreich Wilfried, *Sensor Fusion in Time-Triggered Systems*, Technischen Universitat Wien, Wien, im Oktober 2002
7. Makarov, I.M., Lokhin, V.M. (eds.): *Intelligent Control Methods*. Physmatlit, Moscow (2001)
8. Martchenko Anna, S., Ermolov Ivan, L., Groumpos Peter, P., Poduraev Jury, V., Stylios Chrysostomos, D.: Investigating stability analysis issues for fuzzy cognitive maps. In: *Proceedings of IEEE MED-2003 Conference*, Rhodes, Greece (2003)