Intelligent Data Selection in Autonomous Robot Movement

Olga Bogoiavlenskaia, Dmitry Korzun
Petrozavodsk State University (PetrSU)
Petrozavodsk, Russia
{olbgvl, dkorzun}@cs.karelia.ru

***Abstract*—Autonomous mobile robots have been discovering recently a wide range of applications in various areas of human activities. A robot uses many data sources (sensors) to recognize its current situation, including video, inertial, acoustic, mechanical stress. The data flows from such sources are redundant, error prone, delivered with a rather high rate, and contain considerable information as for the ongoing events so present measurement noise, errors and insignificant data about minor fluctuations of the environment and the device. They present a source information for the inference modules such as navigation, localization, path planning, scheduling etc. The total number of the ongoing events can be large. They have different importance for the movement control. In this paper, we consider a method for intelligent data selection. The method is based on the Additive Increase Multiplicative Decrease (AIMD) algorithm. The effect is in filtering the most significant information to forward to the robot movement control.**

I. Introduction

Autonomous mobile robots have been discovering recently a wide range of applications in various areas of human activities, including industry, commerce, social life, environmental projects, health care, science, education, agriculture, housekeeping [1]. For several decades the efforts of the research community in the area are directed to the development of methods and technologies that could make such devices able to operate in a completely or partly unknown environment without any or with a restricted supervision of a human distant operator, e.g., see [2], [3].

The autonomous movement is subject to Artificial Intelligence (AI) and Ambient Intelligence (AmI) research [4]. Autonomous mobile robot as a system operates in the environment which is either completely unknown and to be discovered or is described in part by a map and a path within the map the robot has to follow. The environment can change during the operation and those changes could be unpredictable. So could do the robot operational mode due to malfunction of unpredictable interaction with the environment. For the proper operation of the system the changes should be detected and a reaction should be designed. For these reasons an autonomous navigation is based upon sensors data which provide varied information about surrounding objects and inner information that could be useful for locomotion and navigation.

In this paper, we propose a fuzzy method for intelligent data selection, which exploits the balance of optimistic and pessimistic expectations of the environment uncertainty. It aims at the estimation of the current rate a sensor delivers essential data. The method is based on the well-known in data communication Additive Increase Multiplicative Decrease (AIMD) algorithm [5], [6]. The intelligence property of the method reflects the human cognition function for the selection problem. The increase or decrease rules are applied when a fault in decision leads to the strong reaction (multiplicative decrease) or a positive sign leads to small reaction (additive increase).

The method is intended to reduce the amount of data passed to processing for the robot movement control, planning and decision making (data inference) modules. The selection is applied before, if possible, or after the raw data smoothing phase. The effect is in filtering the most significant information. As a result, the method advances the situation detection function of a robot by providing high priority information for further recognition and control. This property relates the method to the generic problem of information ranking in AmI [7].

The rest of the paper is organized as follows. Section II introduces the selection problem in sensed data. Section III considers the proposed AIMD based selection method. Section IV describes the particular algorithm to implement the selection method. Section V shows our early numerical experiments to analyze the basic properties of the method. Section VI concludes the paper.

II. Background

Nowadays there exists a wide variety of sensors that deliver diverse multitude of raw data. The sensors could be classified by the nature of the data they perceive, e.g., spatial, temporal, electromagnetic, mechanical and others, or by the nature of the sensor activity, e.g., inner (battery level, wheel angle) external (distance to the obstacle), active (sonar sensor), passive (camera). A range of classifications exists in the literature as well.

The data flows provided by such sources as sensors are redundant, error prone, delivered with a rather high rate, and contain large arrays of the information as for the ongoing events so present measurement noise, errors and insignificant data about minor fluctuations of the environment and the device. Basically, using sensor assumes that further processing of the sensors data could infer useful information about the world around necessary for robot safe navigation and mission completion.

After primary processing the flows are used as input data for the robot movement control subsystems. They present a source information for the inference modules such as navigation, localization, path planning, scheduling, etc. So far there are the following two major problems to solve when interpreting the sensed data.

1. The raw sensors data carry uncertainty due to the lack of precision, external factors intervention or noise of the signal (Gaussian or non-Gaussian).

2. The items in these data flows are not equally significant. Some of them are critical for the device integrity, some are essential for the proper operation and/or mission completion and some are useless.

Let us discuss the second problem more in detail. Single sensor (e.g., camera or inertial sensor) in most cases does not provide sufficient information for control purposes and is not reliable enough. (If it stops the operation the device becomes senseless). So far, the set of sensors carried by a robot normally is vast. Hence the amount of data they produce is as well and the number of events inferred from these data is large. Meanwhile in many cases the attributes enumerated above (critical, essential, useless) could not be attached to a particular sensor. Let us consider a battery level sensor. If the battery level is high there is no need to monitor it often on the regular basis. But if the charge level is low or it is high but decreases faster than expected these data should be examined by the inference modules and future activity should be replanned perhaps.

Hence any sensor can produce crucial or insignificant data but their value depends on many factors and the measurement policy depends on our expectation and the estimate of the environment uncertainty. If the battery is full, we expect the device to operate reliably for some period and there is no need to control it tightly. If the charge level is about to expire, we have to inspect it regularly to plan further moves. Either the device directs to the charging station or in the case of highly important mission follow some other scenario.

To deal with the first problem the data smoothing techniques are widely used. These methods are based in part on the control theory. Different variations of the Kalman filter [8] could be applied for the state estimation if there is a linear low that bounds observed variables, e.g. velocity and acceleration, or distance and velocity. Another major method is Particle filter [9] which is based on sequential Monte-Carlo technique. Due to the computational complexity of these two methods simple techniques as moving arithmetical average, median, mode filters are widely used in practice as well. The sensor producers and the robot designers either apply filters to smooth raw data.

Smoothed data are further processed by sensor fusion algorithms [10], [9], which combine data from several sensors and relevant information from corresponding databases to produce more precise and specific inference about the surrounding environment. Meanwhile Kalman filter and Particle filter could be treated as fusion technique as well, e.g., see [11]. Basically, they are used to evaluate the path traversed for the navigation and localization tasks.

Another approach to data fusion takes more attention recently. Fuzzy methods and intelligent algorithms are applied to events and qualitative properties detection and recognition. In particular, [12] studies neural net for driving style and gait properties identification by the smartphone accelerometer.

Although the first problem enlisted above is researched extensively, meanwhile the second problem is to be solved now by the choice of sensors set and their allocation. This approach cannot reduce the amount of data produced by the chosen sensors. Meanwhile we have two critical restrictions for any autonomous device operation. Those are the battery charge and the computational facilities. Therefore, the implementation of MonteCarlo procedure for every data item (or their essential part), computing matrix operation for Kalman filter or invoking neural network for relatively minor time windows by the device facilities of an autonomous device consumes its operation time. If the data are sent to some server, then wireless data communication consumes battery resources as well. Hence the topical problem we address in this paper is to avoid the fusion of useless data.

III. AIMD Based Data Selection

A. Abstract Sensor Concept

Consider a sensor as a mapping $S_m \mapsto S_n$, where $S_m \subset \mathbb{R}^m$ and $S_n \subset \mathbb{R}^n$. In most cases $|S_m| > |S_n|$ and commonly $m > n.$ as well. These inequalities form one of the main sources of uncertainty for the raw sensor data since the actual mapping is many to one. The set S_m represents the real world phenomenon. It could be continuous or discrete. Also it could be a conjunction of several continuous intervals or be any other subset of $\mathbb{R}^m$. The set S_n is discrete in most cases due to the sensor output granularity. Smoothing raw data done internally by the sensor producer or by the robot computing facilities still keeps it discrete due to the discrete nature of the computing architecture but the latter transforms $S_n \mapsto \tilde{S}_n$ and $|S_n| \leq |\tilde{S}_n|$.

Consider $x, y \in S_n$ and introduce the norm $||x - y|| \in \mathbb{R}$ which defines the distance between two values measured by a sensor. The norm is additive and $||ax|| = a||x||,\ a \in R,\ x \in S_n$. Then let us denote a sequence $s^i = \{s^i_0, s^i_1, \dots\},\ s^i_k \in S_n$ of data produced by ith sensor and a sequence $t^i = \{t^i_0, t^i_1, \dots\}$ of timestamps that label the corresponding elements of the sequence s^i.

Now we have to define the measure of an environment uncertainty which could be used as an identification of significant changes that are worth passing to the data fusion and inference subsystems. Let us consider a single measurement produced by a sensor as a sum

$$||s_k|| = ||\sigma_k|| + r_k,$$

where σ_k is a true value of variable measured and $r_k \in \mathbb{R}$ is an error created by some reason.

If the signal remains unchanged or changes insignificantly there is no need to pass new measurements further repeatedly. Let us assume that $\mathsf{E}[r_k] = 0$. The assumption is natural and means that a sensor is properly calibrated and $\mathsf{E}[s_k] = \mathsf{E}[\sigma_k]$. Now if r_k probability density function is continuous and symmetrical then

$$\mathbf{P}\{r_k > 0\} = \mathbf{P}\{r_k < 0\} = 1/2.$$

Therefore we denote p_m probability that m values of r_n consequently are above or below zero. One could see that $p_m = 1/2^m$. Thus, if the signal changes with probability $1 - p_m$ and remains the same with the probability p_m.

Now let us make an assumption that presence of a sequence in a data stream such that m values of r_k in a row are above (below) zero indicates the presence of the real changes in the environment. Then p_m is the probability of a first kind error. If the r_k probability density function is not symmetrical this normally means that a sensor needs further error alignment and/or calibration. In the case p_m presents upper or lower bound of the first kind error.

In most practical cases value r_k and its expectation are unknown. Let us consider a sequence of m values of s_k increasing or decreasing monotonously. If the signal is constant then this means that r_k is monotonously increasing or decreasing as well. They could be all negative, all positive or both negative and then positive. Hence in the case p_m also could be treated as an estimate for the first kind error probability.

The expectation $\mathsf{E}[r_k]$ could be evaluated using smoothing filters. We propose the exponential average of s^i sequence, since its parameters allow setting of the balance between past and present values in the average and it doesn't require to maintain data windows as, e.g., moving average does. The filter is applied as follows

$$||\hat{s}^i_k|| = \kappa||s^i_{k-1}|| + (1-\kappa)||\hat{s}^i_k||, \tag{1}$$

where $0 < \kappa < 1$.Here κ is the parameter that define the level of the contribution of past and present values.

If for a sensor current value of the exponential average is $\hat{s}^k$ than the estimation of the $\hat{r}_k = ||s_k|| - ||\hat{s}_k||$. Hence the sequence of $\hat{r}_k$ could substitute sequence of r_k.

This additional filter is applied if primary filters don not succeed. Otherwise the primary filter output if available could be used as $\hat{s}^i$ sequence. The filter is not applied for periodical or fluctuating data.

B. The Selection Method

Let us define a delay τ_n, which is applied before the sensor data are passed to the control and inference modules.

The delay is dynamically adjusted depending on the properties of the data in the stream and expresses the relation between optimistic and pessimistic expectations on the environment uncertainty. The delay grows linearly if data do not indicate need for further control and decreases exponentially if the environment may change rapidly.

As we mentioned above the monotonous growth or decrease of the data in the flow may indicate changes in the measured metrics. But there are metrics that normally change during proper operation, e.g., position, sometimes velocity, battery level, video in the stream etc. If this kind of data change according to some expected pattern they could be delayed as well. Therefore, besides increase and decrease values we introduce feedback signals that selection algorithm receives from the control subsystem. The feedback evaluates the significance of the data passed and contributes into multiplicative decrease rate.

The feedback is presented as a sequence $\{k_n\}_{n=0}^{\infty}$, where k_n estimates importance and usefulness of the selected data sent after τ_n interval. The value $k_n \in \mathbb{R}$ and $0 < k_n \leq 1$. It evaluates the importance measure of new information about

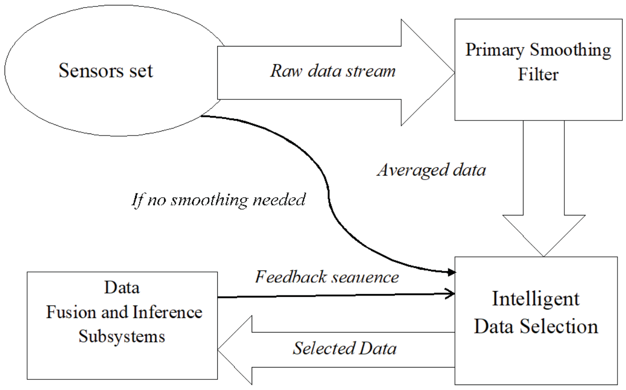

Fig. 1. The sensed data processing

the environment brought to the data fusion and inference subsystems by the new portion of data selected.

In the scale 1 means crucial or very important and 0 means useless. If there is a lack of computing facilities the discrete or even binary value crucial/useless could be applied. Also, we define one special signal k_{urgent} which means that sensors data should be sent immediately.

The general scheme of the interaction between different data processing techniques is presented in Fig. 1.

The flow of the feedback signals generates additional data stream between data fusion system and data selection module but the amount of data is rather small, could be composed out of binary values and if the network capacity is scarce these binary digit could be included in the reserved field of acknowledgment messages of the protocol used.

Hence the delay τ_n evaluates dynamically as follows.

$$\tau_{n+1} = \begin{cases} \dfrac{\alpha}{k_n+1}\tau_n, & \text{if specified events have happened} \\ \tau_n + \delta, & \text{otherwise} \end{cases} \tag{2}$$

Here $0 < \alpha < 1$ is a delay decrease factor or pessimistic level and $\delta > 0$ is the constant that increases the delay or the optimistic level. Therefore, the method reduces the sending rate if the device operation is stable and increases the rate if the environment or the operation mode have been changed. The decrease rate stays within the interval $[\alpha/2, \alpha]$.

The specified events are one of the following

1) The value of $||s^i_k - s^i_{k-1}|| > 0$ or $||s^i_k - s^i_{k-1}|| < 0$ for m time consequently, where m is a parameter.
2) The $||s^i_k|| = (1+\beta)||s^i_{k-1}||$, where $\beta > 0.5$ is a parameter.
3) The value $||\hat{s}^i_k - s^i_k|| > 0$ or $||\hat{s}^i_k - s^i_k|| < 0$ for m times in a row, where m is a parameter.
4) If KF or PF are used the event is: the error evaluated before a correction step does not change sign ν' times in a row.
5) A critical event is identified. The list of the critical events is formed in advance and normally they mean something which needs immediate reaction,

TABLE I. PARAMETERS OF THE DATA SELECTION ALGORITHM WITN NO FEEDBACK.

Parameter	Meaning
α	Multiplicatice deacrese parameter.
δ	Additive increase parameter.
m	The length of meaningful monotonous sequence in the data stream.
InitDelay	Initial value of the delay
UpperLimit	UpperLimit of the delay which shouldn't be exceeded.

e.g., sharp increase of acceleration. Then the signal k_{urgent} is sent.

The first event indicates that the signal value σ_k could change. The second event indicates that the jump in the data is observed and further control signal might be needed. The first even estimates the noise r_n and uses it to indicate the change in the signal. The fourth event follows predefined list which should be formed in advance.

At the end of τ_n interval the current value $\hat{s}_k^i$ is passed to the inference facility.

Therefore, if the sensor data are stable, uniform and correspond the current operation of the autonomous mobile robot they do not contain new insight about the environment and hence are not passed to the data fusion modules. If a noticeable change happens then the data could be significant and are passed to the inference facility for further processing.

If the feedback is not taken into account then (2) transforms into AIMD algorithm as follows:

$$\tau_{n+1} = \begin{cases} \alpha\tau_n, & \text{if specified events has happened} \\ \tau_n + \delta, & \text{otherwise} \end{cases} \quad (3)$$

IV. SELECTION METHOD IMPLEMENTATION

We have developed the algorithm that implements the selection principles described above to a sequence of one-dimensional data $s_k \in \mathbb{R}$. The norm $||s_k|| = s_k$. The algorithm could be applied to a single sensor data that is expressed as a real number e.g. accelerometer, gyroscope, magnetometer and any other real metric including those constructed in advance from the raw data e.g. the distance to a barrier etc.

To test the proposed approach under the hardest conditions we implemented a version with no feedback. The algorithm has no information about the nature of the data and the requirements of the control and planning subsystems and hence relies only on the balance of the optimistic and pessimistic expectations. The algorithm identifies a monotonous sequence of the size m and after the current delay period expires it calculates new value τ_n basing on the expression (3). The version is presented as Algorithm 1.

The sequence identification is not restricted by a single delay period. If the sequence is located across two neighbor periods it will be identified as well. The parameters of the algorithm are described in Table I.

All parameters are integer values. As it was mentioned above the parameters and the relation between them should express the analyst's expectations of the level of the uncertainty and the ability of the algorithm to react on the changes of the environment. Thus if $\alpha > 3$ then the second monotonous sequence in the row will decrease the delay about ten times or more. The initial value expresses our expectation on the reasonable delay duration for the problem at hand and δ affects the ability of the algorithm to recover the delay after decrease. Parameter m could be tuned using the estimates p_m evaluated above. We recommend to set $UpperLimit = 10$ m at least. Otherwise in some cases the delay could become too high and the algorithm could loss contact with the control subsystem.

The algorithm implementation is rather simple so its efficiency is high. All calculations are done on flight by one pass forward. The size of data window to maintain is two which smallest of possible. It does not need to sort values as median filters do. For every data entry two comparison, two addition and two assignments are done. The delay evaluation block needs less than ten operations. The calculation is executed using integer values and if value $\alpha = 1/2$ is used the binary shifts could be implemented instead of integer division. In the algorithm X_i refers to the raw sensor data and S_i to the data selected by the algorithm.

V. EARLY NUMERICAL EXAMPLES

The algorithm was applied to the data set presented in the open free access UCI Machine Learning Repository [13]. We used a data set of single chest-mounted accelerometer. The dataset collected data from a wearable accelerometer mounted on the chest and intended for Activity Recognition research purposes. The data set contains Uncalibrated Accelerometer Data those were collected from 15 participants performing 7 activities. The activities are as follows.

1) Working at Computer
2) Standing Up, Walking and Going up down stairs
3) Standing
4) Walking
5) Going Up Down Stairs
6) Walking and Talking with Someone
7) Talking while Standing

Several researches are devoted to the activities identification problem, e.g. [14]. We chose these data to test the algorithm since they follow the concept the algorithm is based upon. The data present periods of a stable regular activity which switches rarely to another stable activity. Hence the algorithm is expected to filter data within the stability periods.

We applied the algorithm developed to the data presented in the data set. The accelerometer provides values for three axes which were considered separately. The parameters were $\alpha = 2$, $\delta = 2$, $m = 3, 5, 7, 10$ and variable $\mathrm{UpperLimit} = 20$. For x axes the results are presented in Fig. 2 and 3. The stepwise line presents smoothed data where all data within nth delay τ_n are substituted by the data sent to the data inference modules in the beginning of the new τ_n period. The width of the step graphically expresses the length of the corresponding delay. As the algorithm aims τ_n increases when the signal is stable and reaches $\mathrm{UpperLimit}$. If the signal fluctuates then τ_n decreases and reaches its minimum $\tau_n = 1$.

The algorithm reduction facility is one of its key characteristics of, i.e., the amount of data filtered. It could be

Algorithm 1 Real Data Selection Algorithm with no Feedback.

function AdaptiveDataSelectio(α, d, m, $InitDelay$);
{Set initial dealy}
ActiveDelay=τ=InitDelay;
{Initialize the monotonous sequence size counters and the flag}
$inc = 0$;
$dec = 0$;
SeqFound=0;
for $i = 0$ **to** End of data stream **do**
  if $X_{i+1} > X_i$ **then**
    $inc = inc + 1$, $dec = 0$;
  end if
  if $X_{i+1} < X_i$ **then**
    dec=dec+1, inc=0;
  end if
  {Set the flag if monotonous sequence found}
  if $inc > m$ || $dec > m$ **then**
    SeqFound=1;
  end if
  ActiveDelay=ActiveDelay-1;
  if AvtiveDelay=0 **then**
    {Assign new delay value}
    if SeqFound=0 **then**
      $\tau = \lceil \tau\alpha \rceil$;
    else
      $\tau = \tau + \delta$;
    end if
    if τ >UpperLimit **then**
      τ =UpperLimit;
    end if
    {Send current value to the inference subsystem}
    $S_{i+1} = X_i$
    ActiveDelay=τ;
    SeqFound=0;
  else
    $S_{i+1} = S_i$;
  end if
end for
return S;

TABLE II. DATA REDUCTION PROPERTY

	X	Y	Z
$m = 3$	336	277	294
$m = 5$	112	105	169
$m = 7$	67	33	6
$m = 10$	33	27	33

expressed through the total amount of τ_n values assigned. Table II presents this characterizes obtained in the experiments. Total size of the raw data set is 600.

The numerical examples show that parameter m has significant influence on the amount of the data selected. Therefore bigger m values correspond to more optimistic expectations on the environment uncertainty.

For y axes we present the results for $m = 3$ and $m = 5$ in Fig. 4 and 5 to show the influence of relatively small values of m. For $m = 3$ the τ_n value decreases to its minimum and stays at $\tau = 1 \dots 9$ for the data indexes from 156 to 378 and then increases again. For $m = 5$ the delay is low only between 300 and 400.

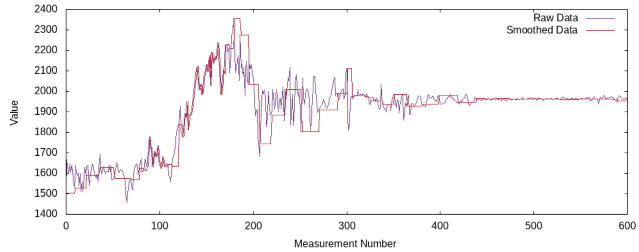

Fig. 2. Smoothing example. X axes. $m = 5$

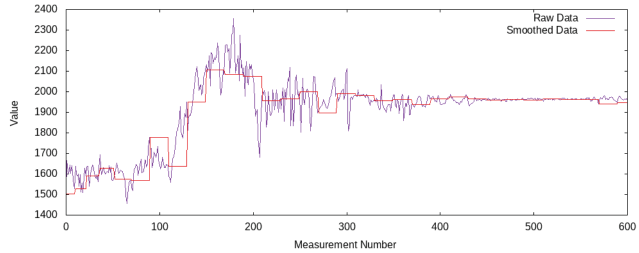

Fig. 3. Smoothing example. X axes. $m = 10$

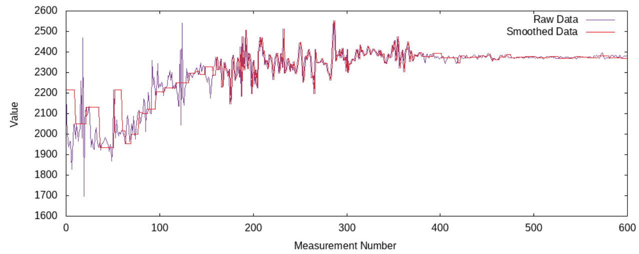

Fig. 4. Smoothing example. Y axes $m = 3$

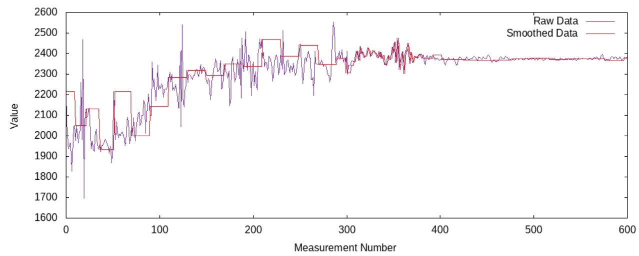

Fig. 5. Smoothing example. Y axes. $m = 5$

For z axes we present the results for $m = 3$ and $m = 10$. For all values of m used in experiments the method removed the outlier at $i = 122$. The results are in Fig. 6 and 7.

Also we evaluated the relative error of the Algorithm in the form

$$\Delta = \max_i \frac{|X_i - S_i|}{S_i}. \quad (4)$$

In all experiments it stays within the interval $10 \dots 15\%$. For z axes the pick at $i = 122$ has relative error about 34% in all experiments as well. If removed from the data set the relative

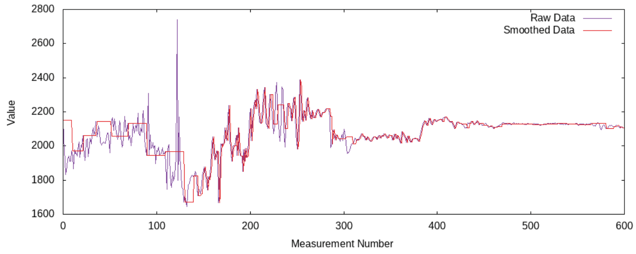

Fig. 6. Smoothing example. Z axes $m = 3$

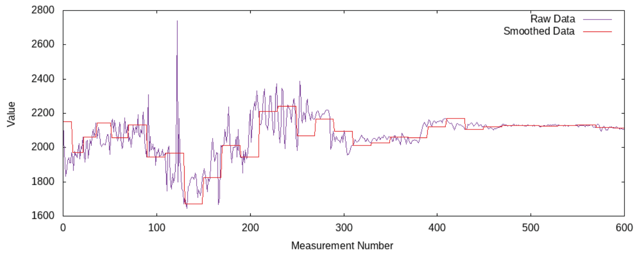

Fig. 7. Smoothing example. Z axes. $m = 10$

error return to the interval mentioned above. So far intelligent selection filters bursts and follows the trend of the data set.

VI. Conclusion

This paper introduced a new fuzzy method of intelligent sensed data selection. The novelty is in using the AIMD strategy, which previously was used in several other problem domains. The method detects the most significant information for autonomous mobile robot movement control, implementing information ranking in large incoming flows of sensed data. In addition, the method employs the feedback exchange between selection algorithms and the sensor data processing. The intelligent selection is close to the rules that human applies in solving a similar selection problem. The method is implemented using the memory saving algorithm with linear complexity. Early numerical examples show the efficiency of the proposed method. We plan to investigate analytical properties of the method, including the estimate of the maximum relative error, the average amount of the selected priority information, and the average delay.

Acknowledgment

This development is implemented in Petrozavodsk State University (PetrSU) with financial support by the Ministry of Science and Higher Education of Russia within Agreement no. 075-11-2019-088 of 20.12.2019 on the topic "Creating the high-tech production of mobile microprocessor computing modules based on SiP and PoP technology for smart data collection, mining, and interaction with surrounding sources". This research is supported by RFBR (research project # 19-07-01027) in part of methods for data-driven edge-centric computing. The work is within collaboration with the Artificial Intelligence Center of PetrSU.

References

[1] S. Zavyalov, A. Kogochev, and L. Shchegoleva, "Robotic and smart service for people with disabilities," in *Proc. 14th Int'l Conf. on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*. IARIA XPS Press, Oct. 2020.

[2] F. Rubio, F. Valero, and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *International Journal of Advanced Robotic Systems*, vol. 16, p. 172988141983959, 03 2019.

[3] R. Barber, J. Crespo, C. Gómez, A. C.Hernámdez, and M. Galli, "Mobile robot navigation in indoor environments: Geometric, topological, and semantic navigation," in *Applications of Mobile Robots*, 2018.

[4] D. Korzun, E. Balandina, A. Kashevnik, S. Balandin, and F. Viola, *Ambient Intelligence Services in IoT Environments: Emerging Research and Opportunities*. IGI Global, 2019.

[5] O. Bogoiavlenskaia, A. Vdovenko, D. Korzun, and A. Kashevnik, "Individual client strategies for active control of information-driven service construction in IoT-enabled smart spaces," *International Journal of Embedded and Real-Time Communication Systems (IJERTCS)*, vol. 10, no. 2, pp. 20–36, 2019.

[6] M. Allman, V. Paxson, and E. Blanton, "TCP congestion control," in *RFC 5681, September*, 2009.

[7] D. Korzun, A. Voronin, and I. Shegelman, "Semantic data mining based on ranking in internet-enabled information systems," in *Fuzzy Systems and Data Mining V — Proceedings of FSDM 2019, Kitakyushu City, Japan, October 18–21, 2019*, ser. Frontiers in Artificial Intelligence and Applications, A. J. Tallón-Ballesteros, Ed. IOS Press, 2019, vol. 320, pp. 237–242.

[8] W. van Drongelen, "Chapter 19 - kalman filter," in *Signal Processing for Neuroscientists (Second Edition)*, second edition ed., W. van Drongelen, Ed. Academic Press, 2018, pp. 361–374.

[9] Y. Raaj, A. John, and T. Jin, "3d object localization using forward looking sonar (fls) and optical camera via particle filter based calibration and fusion," *OCEANS 2016 MTS/IEEE Monterey*, pp. 1–10, 2016.

[10] K. Nagla, M. Uddin, and D. Singh, "Multisensor data fusion and integration for mobile robots: A review," *IAES International Journal of Robotics and Automation (IJRA)*, vol. 3, 09 2014.

[11] A. T. Erdem and A. Ö. Ercan, "Fusing inertial sensor data in an extended kalman filter for 3d camera tracking," *IEEE Transactions on Image Processing*, vol. 24, no. 2, pp. 538–548, 2015.

[12] C. MacAdam, Z. Bareket, P. Fancher, and D. Ervin, "Using neural networks to identify driving style and headway control behavior of drivers," *Vehicle System Dynamics - VEH SYST DYN*, vol. 29, pp. 143–160, 01 1998.

[13] Uci machine learning repository. center for machine learning and intelligent systems. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Activity+Recognition+from+Single+Chest-Mounted+Accelerometer

[14] P. Casale, O. Pujol, and P. Radeva, "Human activity recognition from accelerometer data using a wearable device," vol. 6669, 06 2011, pp. 289–296.