Agronomy Research **17**(4), 1688–1704, 2019
https://doi.org/10.15159/AR.19.140

Measurement Robotic Arm (MRA) for the evaluation of localization sensors properties

P. Matějka[1,*], J. Kadeřábek[2] and V. Shapoval[2]

[1]Czech University of Life Sciences Prague, Faculty of Engineering, Department of Technological Equipment of Buildings, Kamýcká 129, CZ16500 Prague, Czech Republic
[2]Czech University of Life Sciences Prague, Faculty of Engineering, Department of Agricultural Machines, Kamýcká 129, CZ16500 Prague, Czech Republic
[*]Correspondence: matejkapavel@tf.czu.cz

Abstract. The purpose of this longitudinal project is to verify the dynamic properties of the Real Time Kinematic receivers ('RTK'). For this purpose, a verification method using Measurement Robotic Arm ('MRA') described in this paper has been developed. This device can be moved along a circular trajectory in a horizontal surface. Using the methodology described in this paper, the absolute position of the MRA trolley in absolute time can be defined with absolute accuracy and can serve as a reference for the verification of RTK receivers positioning. Its movement, including its breaking, can be controlled via a PC app and various sensor properties can be monitored. The position of the trolley is determined by the encoder. A Hall sensor indicates the absolute start position of one full turn. The absolute time marks of the measurement and the time synchronization of the microprocessor based on Pulse Per Second ('PPS') were obtained from the standard GNNS receiver. This study provides information about a proposal solution of the MRA reference system in terms of the frame construction description, the design of electronic equipment and the design of a software solution for processing and logging of messages. This paper also presents the results of three tests performed to verify the functionality and reliability of the MRA system: PPS time accuracy verification, the quantity and correctness of sent messages according to the arm instantaneous speed, and a real RTK verification test. The MRA also can be used to verify the dynamic properties of other localization devices.

Key words: localization, accuracy, precision, dynamic properties, reference system, reference trajectory, time synchronization, PPS, RTK, GNSS, agriculture, etc.

INTRODUCTION

The purpose of this study was to build and verify the equipment for evaluating the dynamic properties of localization systems. The main task was to develop a new method to evaluate the dynamic properties of the Real Time Kinematic ('RTK') receivers during the 'fix' state when the RTK receivers can measure most accurately. The RTK is the most accurate localization technology among Global Positioning Systems (GPS) that is available on the market. This method uses a correction signal from reference stations for more accurate positioning (Feng & Wang, 2007).

Thanks to its accuracy of around 20 mm, this localization method can be applied in many industries, e.g. in precision agriculture for tractor guidance. Tests of evaluating the

dynamic properties of the accuracy of localization systems have already been performed. In one study, tractors were moved in a defined path at speeds ranging from 0.83 to 1.94 m s^{-1} and the deviation ranged from 20 to 30 mm (Gan-Mor, 2007; Carballido, 2014). Other studies tested field robots. For instance Bakker et al. (2011) has shown that during a field test, where a robot followed a strait path at speed of 2 km h^{-1}, the lateral error was from 16 to 45 mm. An extremely successful measurement by Jilek (2015) has shown that lateral errors were less than 10 mm when the robot followed a sequence of waypoints. Unfortunately, in any of these measurements, time synchronization was not taken into account. In other studies, where the time deviation was defined, the precision of more than 10 mm s^{-1} was obtained for the instantaneous speed estimate provided by a consumer-grade GNSS (Global Navigation Satellite System) receiver (Boffi, Gilgien & Wieser, 2016). In this research, a downhill coaster track with a set of photocells and antennas placed on a coaster car was used as reference system.

Another method for evaluating the receiver's accuracy is processing the data from localization sensors as deviation from a reference point, RMS error or a number of satellites. This data has been evaluated in several studies through static or dynamic measurements for evaluating the accuracy and precision of localization sensors (Feng & Wang, 2008; Garrido et al., 2011; Berber et al., 2012; Kabir et al., 2016). This methodology was also used in authors' previous research where the RTK receiver's capability to determine the accurate position was evaluated through static measurements (Kadeřábek et al., 2018). However, the aforementioned method is more suitable for static measurements and, in case of dynamic movement, the determination of reference points changing in time during the movement is problematic. For dynamic measurements, a precision device in range of mm capable to provide location data at certain time points is needed as reference equipment. The purpose of this research was to build such equipment. Longitudinal tests had been conducted and a prototype of the equipment was built. It was then decided to build a Measurement Robotic Arm ('MRA') that would move on a circular trajectory which is easier to monitor in time positions. This method was chosen because of its high expected accuracy and minimal technical requirements (on the mechanical construction as well as on the conduct of measurements themselves).

This study focuses on the design and development of the methodology for future validation of RTK receivers. The MRA can also be used in future to validate other localization systems, such as incremental systems (systems using data fuse of Inertial Measurement Unit sensors, vehicle odometer systems, the Correvit laser sensor, the mouse-based camera sensor, etc.) and absolute measuring systems (based on the method of triangulation or trilateration). Furthermore, it also allows for the verification of systems that are used for object tracking based on a machine vision technology: laser scanner systems, camera systems, ultrasonic systems, infrared sensors or the fusion of them all). Moreover, it is also possible to use the MRA during the development and tuning of localization algorithms. Raskaliyev et al. (2017) used similar reference system for the development and debugging of algorithms that process information from IMU. The MRA can also be used as a reference system for machine learning development annotation process.

This study provides information about a proposal solution of the MRA reference system in terms of the frame construction description, the design of electronic equipment (namely its variant for RTK tests) and the design of a software solution for processing and logging of messages. This paper also presents the results of three tests performed to

verify the functionality and reliability of the MRA system: Pulse per Second (PPS) time accuracy verification, the quantity and correctness of sent messages according to the arm instantaneous speed, and a real RTK verification test demonstrating the future benefits of this longitudinal project.

MATERIALS AND METHODS

The system of Measuring Robotic Arm ('MRA') consists of three main parts: the frame construction with sensors, motors and brake system, the hardware part with all main electronics (microprocessors, data loggers, operation centre, voltage source, etc.) and the software part providing for a distant controlling of the MRA and processing of sensors signals. The parts are described in the following chapters.

MRA construction

The MRA (Fig. 1) was made of aluminium prototype extrusion profiles 30 mm and 40 mm wide that are widely used in different industries. The centre part of the rotation was static and consists of a cross with main axis of the rotation. The aluminium cross was attached to the ground to provide stable and repeatable placement of the MRA during all measurements and at all tested speeds. A rotary slip ring KS 54-D12-12x5A was placed on the main axis to supply electricity for all systems during the rotation. The axis was connected to the arm with industrial bearing. This bearing connected a movable part of a trolley and a beam with the axis. A cogged pulley was attached to the axis to transfer the rotary movement via a cogged belt to Solid-Shaft Incremental Encoder ZSP5208-001G-2500BZ3-5E that was placed on the beam.

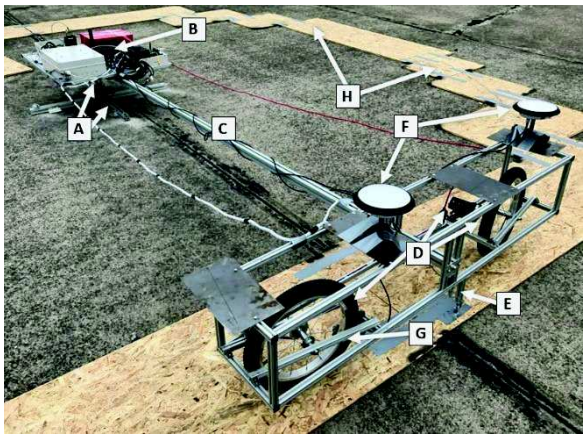

Figure 1. The Measuring Robotic Arm set during measurement. (A – cross with the rotation axis; B – board with electronics attached to the beam; C – beam; D – trolley with wheels and motors; E – Hall sensor; F – two antennas; G – brake; H – OSB boards).

The board with electronics was placed above the static axis and attached to the beam. It was placed in the centre of rotation of the construction. The main purpose was to concentrate all the heavy electronics in the centre of rotation to limit a rapid increase of the centrifugal force. This force can affect the accuracy of measurements or

destabilize the construction at a high speed which could result in a collision and damage the expensive equipment.

The trolley, connected to the beam, enabled to assemble four antennas in 0.5 m distance from each other and at 3 m distance from the axle, the centre of rotation, along a circle line. The trolley was moving on two 16-inch wheels that placed on opposite sides at 1.22 m distance of to ensure better stability. These wheels helped to move the trolley smoothly and reduced vibrations that can affect measurements. Both wheels are driven by DC motors with 100 W power each. The motion power system (motor driven wheels) was placed on the trolley with the goal to eliminate small bending or deformation of the construction during acceleration, deceleration or movement. These drive forces did not burden the construction on the beam, axis or trolley during the movement, and thus, an overall better accuracy of the system was ensured.

The cuboid shape of the 0.44 m high trolley construction including wheels enabled to place the antennas above the construction in order to eliminate possible signal disturbances. The trolley was equipped with one V-brake type brake driven by a linear DC motor for the purposes of evaluating the properties of sensors during the deceleration or stopping the movement after each measurement. The Hall sensor was placed on the bottom side of the trolley, close to a magnet fixed on the wooden OSB boards. The aim of was to ensure that the sensor records each full turn of the MRA.

MRA electronic equipment

The electronic equipment consists of several components that, as a whole, provide required functionality of the MRA and communication between each component. Fig. 2 describes a component linking model, namely a variant for verifying the dynamic properties of up to four RTK receivers using one channel of the VRS correction via the Internet. However, the MRA was developed as a variable system, which means that its components can be exchanged to verify the dynamic properties of other localization systems. The core of the platform is the MRA embedded system, which consists of the Sensor Board ('SB'), Motor Control Board ('MCB') and Break Control Board ('BCB').

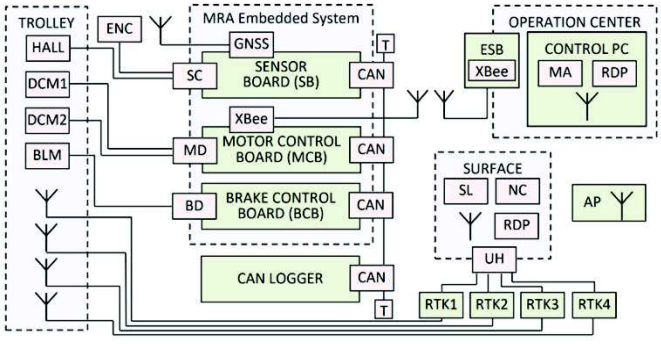

Figure 2. The connections scheme. (MD – motor driver; BD – brake driver; DCM1 and DCM2 – DC motors; BLM – brake linear motor; SC – sensors connector; CAN – CAN bus module; T – terminator resistor; UH – UART/USB hub; AP – cellular modem with WiFi connection; MA – MRA Application; RDP – software using Remote Desktop Protocol; SL – serial logger (Putty bash script); NC – Lefebure software as NTRIP client; ESB – Emergrency Stop Box).

The SB unit is used to process signals from the MRA sensors (according to the algorithm described in the subchapter below). It creates records in the form of messages sent over the CAN bus. The hardware of this unit is based on the Arduino DUE single-board computer platform with the ATMEL SAM3U microprocessor. The SB is directly connected to the Adafruit Ultimate GPS Breakout GNSS receiver via the UART interface, and the SC sensor connector is the output. The Proximity NJK-5002C Hall Sensor ('HALL') and Encoder ZSP5208-001G-2500BZ3-5E ('ENC') are connected via the SC to the digital inputs of the SB. The power supply for the HALL and the ENC is also powered via the SC. As an interface for the CAN bus, the MCP2515 is connected directly to the SB via the SPI interface. The MCU is based on the Arduino DUE platform (as well as the SB) and serves primarily to control the MRA engines. Two 100W/24V DC motors (DCM1 and DCM2) are controlled by the Pololu Dual VNH5019 driver shield motor module and are located on the MRA trolley. In addition, this unit serves as the communication interface of the CAN bus (again via the MCP2515 module) and the wireless connected Operation Centre. This part is a kind of 'communication node' of the entire MRA system. The Xbee-S2C module with 2,4 GHz wireless communication via the UART interface was used to communicate with the Operation Centre. The BCB is only used for controlling the braking system. It is based on the Arduino UNO single-board computer with ATmega328P microcontroller. The MCP2515 module in the CAN-BUS Shield V1.2 variant was used to communicate with the CAN bus BCB. The braking mechanism is controlled by a 4-channel relay of the DSP AVR MSP430 which operates a 12 V DC linear motor with an axial thrust of 1,300 N.

Two logging devices were used for recording the measurement process. The first logger ('CAN LOGGER') was designed to read and store CAN bus messages sent by the SB. For this purpose, a high-end automotive logger Vector (GIN) GL4000 was used. However, the MRA platform was designed in a way that enables to replace this logger in future by any comparable CAN bus logging device. The second logging device ('SURFACE') was designed to record the output communication of the RTK receivers (in this case up to four). These outputs were assumed to be the UART communication standard via which the measured positions and time stamps were sent from RTK receivers as NMEA messages. SURFACE was realized by the Microsoft Surface Pro 4 laptop (CPU Intel i5 7300U, RAM 8GB, USB 3.1, Windows 10x64). The AXAGON HUE-S2B-4 USB 3.0 ('UH') was connected to the SURFACE USB interface and individual RTK receivers were connected to the UH. However, SURFACE served not only as a logging device but also for the purpose of assuring the Networked Transport of RTCM via Internet Protocol ('NTRIP') client. By this Trimble, VRS Now correction signal ('VRS') can be distributed to all RTK receivers via the UART interface. As an Access Point ('AP'), mobile TP-LINK M7350 Wi-Fi Modem was used with the support of 3G/LTE (800, 900, 1800, 2100, 2600 MHz) and Wi-Fi 802.11a/b/g/n standards with 2 Ah battery.

The MRA can be controlled from the Operation Centre which was implemented in a laptop Lenovo IdeaPad 720s-13IKBR (CPU Intel i7 7300U, 8GB RAM, SSD 256GB, USB 3.0, Wi-Fi 802.11ac, Windows 10x64). The Emergency Stop Box ('ESB') was connected to this laptop via the USB/UART converter. The EBS includes the Xbee-S2C wireless communication module. A shredding button was placed on the UART communication interface cable and directed from the Operation Centre to the MRA. This button can be used to stop the movement of the operating MRA. The Operation Centre

was connected to SURFACE via the AP with using Remote Desktop Protocol ('RDP'). This connection was used for the purpose of managing the RTK receivers, including logging and setting up a NTRIP client.

MRA software solution

The software solution is based on several separate mutually cooperating units that provide the control of the MRA and processing of its sensor signals (GNSS, Hall sensor and encoder). The most important software unit is the firmware installed on the Sensor Board ('SB'), which is based on the Arduino DUE development platform. The code for this unit was written in the Wiring programming language version 1.0. It was optimized in order to assure the highest possible quality and signal processing speed for three sensors using Arduino DUE board hardware interrupts. The optimization was supported by the DueTimer.h library version 1.4.7. Some message losses happen at higher sensor pulse quantity due to the speed of rotation of the encoder. These message losses did not however cause the loss of the arm position data because this information was stored in the SB internal memory. The validation of this and the verification of the quantity of sent messages by the SB unit according to the arm's rotation speed will be described below.

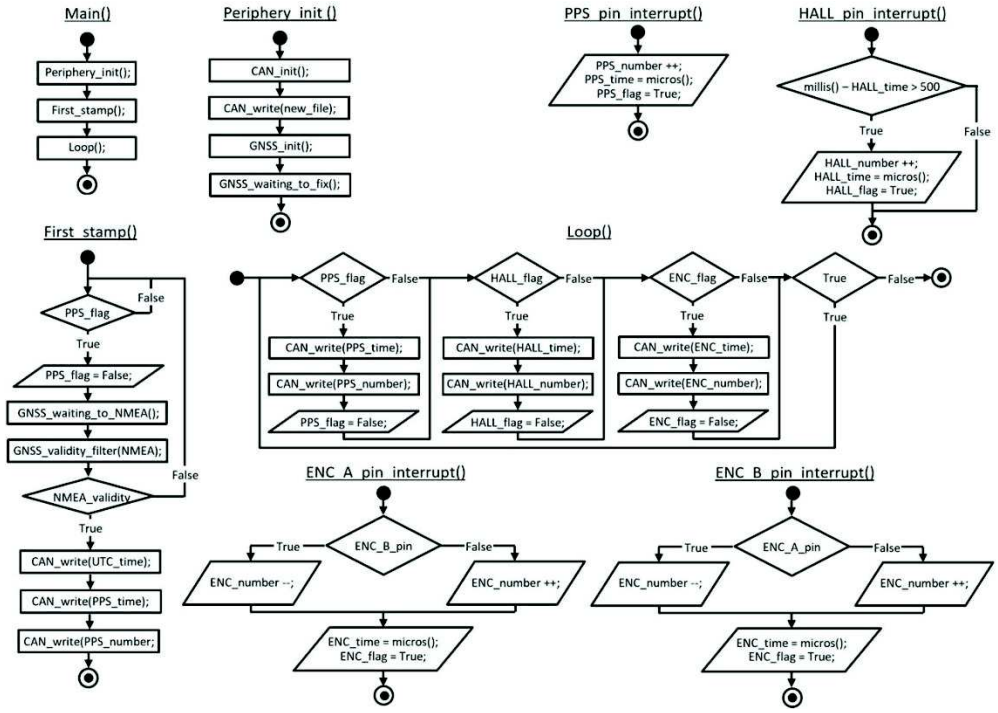

Figure 3. UML Activity Diagram of SB algorithm including interrupt headers of three sensors.

The simple principle of signal processing method of the SB unit is described in the UML (Unified Modeling Language) Activity Diagram (Fig. 3). The process begins with the Periphery_init() method in base branch Main(). In the CAN_init() method, CAN bus communication is initialized using the mcp_can.h library version 1.1.1 at the speed of

500 kb s^{-1} with 11-bit identifier. Next, the CAN command is sent as a message for generating of a new measurement file at the CAN LOGGER. Next, the GNSS_init() method initializes the initial state of the measurement. An internal GNSS is set up in a proper communication mode on the serial line (115,200 kb s^{-1}, 8, 1, N, GPGGA, 1 Hz). Then, the algorithm goes into a loop in method GNSS_waiting_to_fix() until the signal state of the GNSS 'FIX_state' is indicated. The goal of next main method First_stamp() is to determine the absolute alignment of the starting time of measurements. In the first loop, the algorithm waits for the interrupt flag set in PPS_pin_interrupt(). If this signal occurs, a NMEA message on the serial port is expected by method GNSS_waiting_to_NMEA() and its validity is subsequently tested by the filter GNSS_validity_filter(NMEA). The filter must fulfil two conditions: the NMEA report meets its checksum and the time stamp contains an undamaged value. If the filter is valid, the time stamp in UTC format, included in a valid NMEA message, is sent to the CAN bus. Also both values obtained in interruption are sent to the CAN bus. In the method Loop(), it is checked whether there an interrupt flag was set by at least one of the four interrupts. Four interrupts secure three sensors (PPS, Hall sensor and encoder) because the encoder must be secured by two interrupts. In the positive case of at least one set flag, the obtained information (microprocessor time and the iterator of the interrupt) are sent to the CAN bus. Each type of message of each sensor which is sent on CAN bus has its own unique address. Individual interrupt processes are also shown in Fig. 3.

The MRA Application ('MA') was developed in the C# programming language using the. NET framework version 4.6.1. This application allows to control and to check the individual functions of the MRA via wireless transmission. Its primary function is to control the MRU motion with the MCB unit. It is possible to set parameters for the MRA instantaneous speed and acceleration. The MA also allows to restart the SB unit for start of a new measurement file in CAN LOGGER, as described Fig.3 It also allows to control the brake by the BD unit when it is necessary to influence the measuring trajectory or to stop the trolley movement in emergency. Furthermore, the MA allows the operator to read the interruption indications from the individual sensors and to show the approximate angular speed of the trolley.

The firmware for the MCB and BCB units, which is the same as for SB unit, was written in the programming language Wiring version 1.0. The task of the MCB was to ensure a two-way communication with the MA and a two-way communication with the devices connected to the CAN bus. The MCB enables to control the motor driver, including a watchdog feedback protection (signal sending from the MA to the MCB) against dangers caused by a failure of wireless signal. The BCB firmware enables to control the trolley's brake mechanism.

Verification of the time accuracy of the PPS signal

The first test was based on a general verification of the time accuracy of the PPS signal (Pulse per Second). A similar work is discussed in the study of Niu et al. (2014), which takes other types of GNNS receivers into account, and uses the reference system in the form of atomic clocks to verify the time accuracy. Our study looks at this verification from the point of view of mutual verification between PPS signals of three GNSS receivers using statistical verification method. For this to be done, the selection of the GNSS receiver is essential to align the time base of the SB unit. If the microcontroller clock of the SB unit is checked regularly (one time per one second), a correction of

deviations can be made in the subsequent post-processing. Thanks to using of GNSS in the MRA it is subsequently also possible to provide accurate absolute time alignment of two signals (from the encoder and the Hall sensor) that ultimately determine the position of the arm during the whole measurement period. The following GNSS receivers have been tested: Adafruit Ultimate GPS Breakout, U-blox C94-M8P and Garmin 18x LVC.

The logic analyser Saleae Logic 8 was used to log the PPS output signal from three GNNS receivers, with a set sampling frequency of 25 MHz. To log the data in Lenovo E540 (Intel i3-4000M CPU, 8GB RAM, USB 2.0, Windows 10x64 OS), the Saleae Logic logging software version 1.2.18 was used. The GNSS receivers were connected to power supply and three connected identical GNSS antennas FURUNO AU-15 were placed on a metal plate of the trolley in a stable position with an open access to the orbit. The GNSS receivers were set up to receive only GPS signals. Measurements took place at the location of 50°04'32.8"N, 14°31'10.3"E and took 3,600 s. The screenshots (Fig. 4) from Saleae Logic logging software show the cut of the measurement of three PPS signals.

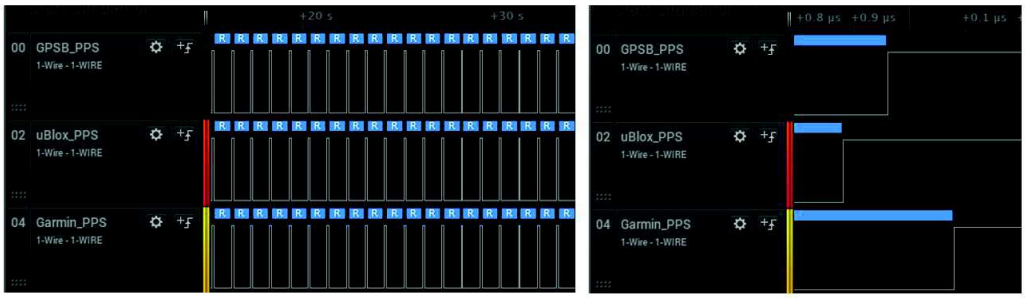

Figure 4. Screenshot of Saleae Logic logging software from the measurement of three PPS signals.

The measured data was processed in the Spyder software version 3.2.8 as part of Anaconda distribution version 5.2 that used the Python programming language version 3.6.5. The goal was to calculate the deviations of the individual upcoming edge of PPS from the GNSS receivers against the clock of the logic analyser. This calculation is described by formula (1):

1. PPS time errors (Δt_{seleae}) – data-set of time deviations of PPS opposite to logic analyser clock:

$$\Delta t_{seleae_i} = t_i - t_{i-1} - 1 \qquad (1)$$

where: t_{pps} – value of measured PPS sample by logic analyzer; i – iterator of PPSs DataFrame, $i \in \langle 1; 3{,}599 \rangle, i \in N$.

Verification of the quantity and correctness of the sent messages according to the arm instantaneous speed

The second verification aimed at verifying the ability of the MRA to process signals from the sensors and to send them to the CAN bus. This test was performed with the help of the MRA device. The RTK receiver Tersus BX-305 was also attached to the MRA, but the verification of this sensor is not taken into account in this study. The MRA was assembled, anchored to a flat concrete surface, and OSB boards were installed under

the trolley's trajectory. Also, the MRA sensors (encoder using a cogged belt to both pulleys, Hall sensor with a magnet and SB internal GNSS receiver antenna) were installed, calibrated and verified. During the measurement, fifteen tests were performed with an approximate duration of 30 seconds at different speeds (with max. speed 8.3 m s^{-1}). Before each measurement, the SB was reset in order to create a new measurement file. Each measurement ended by stopping the trolley by brake.

Thee measurements were evaluated by using the developed Python script (using the same components described in the previous chapter). Because of the largeness, complexity and variability of the calculations, all individual steps will not be described. The important calculations of time corrections are specified below. Other calculations are based on basic knowledge of physics, geometry and data processing methods.

The first task of the script was to parse the messages in the CAN messages file stored on the CompactFlash card. The messages are sorted into variables of DataFrame data structure from the Pandas library version 0.23.4 according to the sensor type (PPS, Hall sensor, encoder). From these structures, the values of the microprocessor time t_{fix} with the t_{nmea} were extracted. The t_{fix} value represents the last stored value of the microprocessor time of the first interruption triggered by the leading edge of the PPS signal when the valid NMEA message passed through the filter (see the MRA software solution chapter). The t_{nmea} value represents this valid NMEA message converted from the UTC HH:MM:SS.sss time format to the float format of the running total of seconds in a day starting at midnight (time of day in seconds).

This pair of values guarantees the measurements data in absolute time. It was necessary to subtract the value t_{fix} from all the measured times of all sensors t_{pps}, t_{hall} and t_{enc}, as at this time the SB unit started the measurement. In addition, the clock deviations of the SB microprocessor from the time of the measurement were calculated by the following formula (2):

2. Sensor board time error difference (Δt_{err_i}) – data-set of time deviations differences of Sensor Board clock opposite to the PPS signal:

$$\Delta t_{err_i} = t_{pps_i} - t_{pps_{i-1}} - 1 \qquad (2)$$

where t_{pps} – saved values of microprocessor-time by Sensor Board to CAN messages in times of PPS interrupts with offset of start of measurement; i – iterator of PPSs DataFrame, $i \in \langle 1; number\ of\ PPS\ messages \rangle, i \in N$.

The set of obtained deviations Δt_{err} from one selected measurement (which is further used as example) was shown in Fig. 5. Furthermore, this set of errors Δt_{err_i} for time marks correction used the Hall sensor t_{hall} and the encoder signal t_{enc} with the help of relations (3), (4):

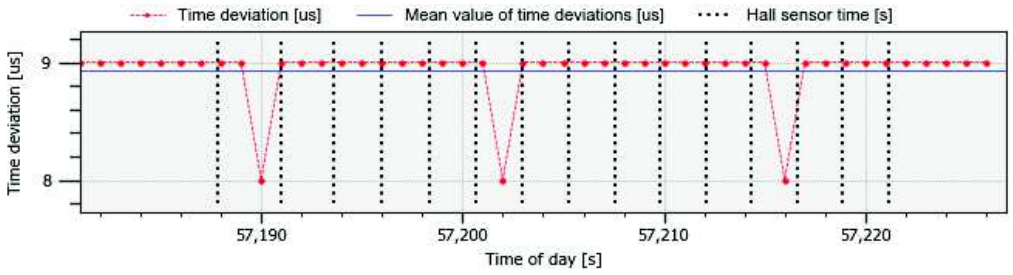

Figure 5. System Board time deviation in time.

3. Sensor board time error (t_{err}) – data-set of time deviations of Sensor Boards clock opposite to the PPS pulse:

$$t_{err_i} = \sum_{j=1}^{i} \Delta t_{err_j} \tag{3}$$

where Δt_{err} – data-set of time deviations differences of Sensor Boards clock opposite to the PPS signal; i – iterator of PPSs DataFrame, $i \in \langle 1; number\ of\ PPS\ messages \rangle, i \subset N$; j – iterator of summation, $j \in \langle 1; i \rangle, i \in N$.

4. Hall sensor or encoder corrected time (t_{enc_corr}) – data-set of time corrected time-stamps of the Hall sensor or encoder signals:

$$t_{enc\_corr_i} = t_{enc_i} + t_{err_{int(t_{enc_i})-1}} + \Delta t_{err_{int(t_{enc_i})}} * \left(t_{enc_i} - int\left(t_{enc_i}\right)\right) + t_{nmea} \tag{4}$$

where t_{enc} – saved values of microprocessor-time by the Sensor Board to CAN messages in times of encoder interruptions with the offset of the start of the measurement; t_{nmea} – value of the first valid string of NMEA message providing absolute placement in time; i – iterator of encoders DataFrame, $i \in \langle 1; number\ of\ encoder\ messages \rangle, i \in N$.

For the interpretation of formula (4), an encoder corrected time formula was selected. This code was also used for calculating t_{hall_corr} (Hall sensor corrected time). Now, when all times from both sensors were corrected, it was possible to start off assigning individual messages from the encoder's appropriate angle of the MRA. This was performed by algorithmically scanning the encoder times t_{enc_corr} and assigning them the value of a particular angle of turn according to the Hall sensor time t_{hall_corr}, which was used for this purpose as a boundary value. The value of the interrupts order of the encoder then conveys, together with the knowledge of the number of encoder pulses per turn, the specific angle of the arm rotation. Next, it is necessary to calculate the offset of the angle from the first turn before the first pulse of the Hall sensor. With the knowledge of the angles and the knowledge of the initial absolute position of the MRA (given from two surveyed points), it is possible to calculate with simple goniometric functions the relative coordinates of the MRA trajectory in the Cartesian coordinate system (for example East-North-Up or East-North-Down) (see next subchapter). It is also possible to obtain the measured trajectory in an absolute representation in a geographic coordinate system (such as WGS84). The calculated positions, for data analysis purposes, can be limited to two-dimensional view depending on the time (Fig. 6). These positions can be used as the final reference points forming the reference trajectory.

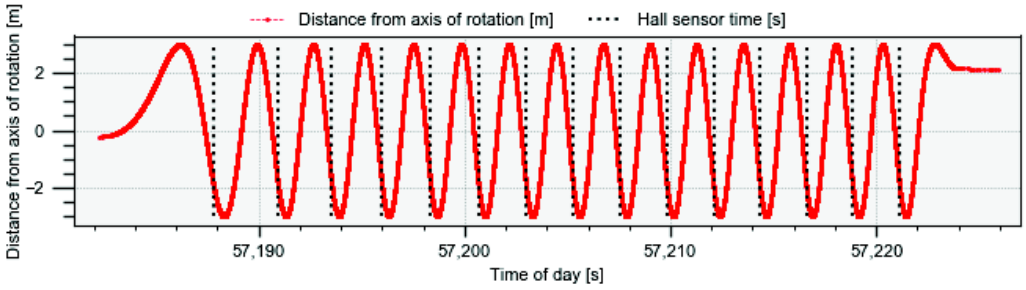

Figure 6. The relative positions of 'x' axis of the MRA trolley in time.

For further analytical processing of the results, such as verifying the quantity of sent messages according to the instantaneous speed, it was necessary to know the instant angular velocity of the MRA. This was obtained from the difference of angular distance and time between the two calculated points of trajectory (Fig. 7). However, since some messages were omitted (see the results in this chapter), it was necessary to eliminate this destroyed signal by deploying the de-noising technique Savitzky-Golay smoothing and median filtering as used by Mishra, (2019). Since we can assume that the trolley has a high value of inertia at the tested speeds (to 8.3 m s^{-1}), we can consider the distortion of this filter to be minimal.

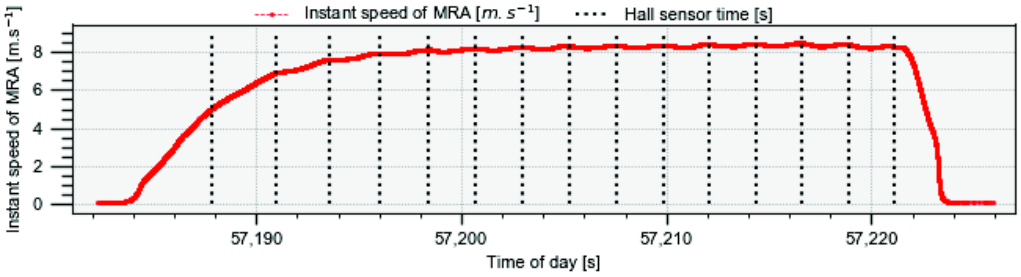

Figure 7. The instant speed of the MRA trolley in time.

Measurement of the RTK system

The Tersus BX-305 RTK system was measured to demonstrate the benefits of this project. The processing of the results in this paper did not include the evaluation of specific values of the RTK receiver. The aim was, with the help of visualization, to show the ability of the system to determine the exact position in different modes.

The measurement took place at an old military airport runway in Milovice in Nymburk District, Czech Republic on the 1st of November 2018 (50°14'9.1"N, 14°55'22.4"E). This location was chosen because of the flat horizontal concrete surface without high buildings, trees or other obstacles in the near surroundings. The measurements were part of the measurement set used in the evaluations of the previous subchapter. As a reference signal, the VRS signal was chosen. A virtual base station was located at the centre of the MRA. The measurements were made in three RTK receiver modes. The first was a non-correction mode, the second used the VRS correction, and the third used the VRS correction with the built it fusion of internal IMU with this RTK receiver. Each measurement was carried out by rotating the MRA at three different speeds (2.7, 5.5 and 8.3 m s^{-1}) for about 30 seconds and then the movement was stopped by brake. The last measurement at the speed of 8.3 m s^{-1} with the activated internal IMU and VRS correction were shown in the subchapter above (Figs 6 and 7).

For future location of the reference trajectory to the absolute space, it is necessary to determine the location of the centre of the axis of the MRA and its inception angle of rotation. This will be done by one of the RTK receivers that will survey two points using the geodetic method as in the case of study of Gao (2011). The first point will be measured at the point of the axis placement, the centre of rotation of the MRA (after MRA dismantle). The second point will be measured anywhere on a straight line that intersects the centre of rotation (axis) and a centre point of the trolley at a distance of

30–50 m. At this MRA position the Hall sensor pulse detection point will be located. To determine the surface tilt of the OSB boards 8 more points will be carried out at the circular trajectory of the trolley.

RESULTS AND DISCUSSION

Verification of the time accuracy of the PPS signal

Three sets of 3,600 samples of data were obtained and processed according to formula (1) from the three GNSS receivers. These three sets of deviations in microseconds were plotted in Fig. 8, where the horizontal axis represents the order of the samples and the vertical axis represents their deviation Δt_{seleae} against the clock of the logic analyser in microseconds. It was clear that these deviations of all three GNSS receivers have a very close relation, as it was evident from one of the represented segments of the measurement (Fig. 8). The time deviations of all three PPS signals in this measurement showed very similar values in maximum and average values. In both cases were these deviations in order of microseconds, concretely the maximal value $\Delta t_{seleae_{max}} = 5.7201\ \mu s$ and the mean value $\Delta t_{seleae_{mean}} = 4.693\ \mu s$.

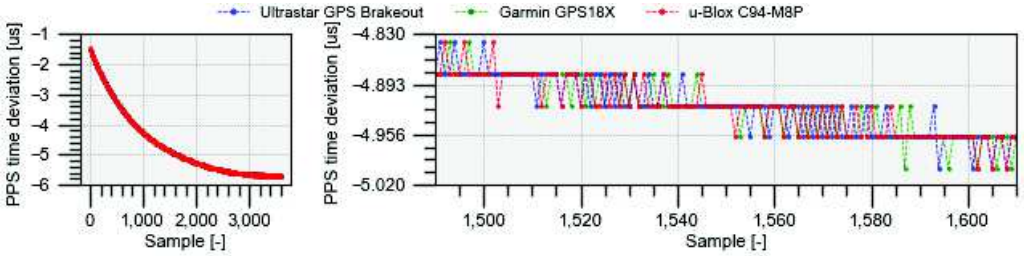

Figure 8. The time deviation of three PPS signals against the logic analyzer clock.

Since the paper of Niu et al. (2014) showed the deviations of the PPS signal of GNNS receivers against the measuring device with atomic clocks as the reference time source (working with systematic error of 0.25 ns) in tens of nanoseconds, it was assumed that the error occurred in our measurement method. Furthermore, it was evident from the deviations of the three PPS signals that the error has the same gradually decreasing non-linear trend. This may be caused by the deviation of the time base (i.e. the internal oscillator of the logic analyser), for example due to the temperature in the outside measurement area or many other negative influences on this equipment (Zhoul et al., 2008).

The objective of this study was to verify whether a particular type of GNSS receiver can be used as an undivided component in order to assure a time base functionality of the MRA. Since insufficient time stability of the logic analyser clock was detected and it was not to possible to bring originally intended results, it was useful to at least verify the PPS signal and in particular to detect whether the three sets of time deviations of the PPS were statistically significantly different or not. The histogram (Fig. 9) showed the distribution of the measured PPS signals deviations against the logic analyser clock in microseconds. Since the data of the PPS time deviations were not distributed normally, it was not possible to use the Analysis of Variance parameter test. Therefore,

nonparametric Kruskal-Wallis test was chosen. The p-value = 0.999 of this test was significantly higher than 0.05, so the zero hypothesis was not rejected. Scattering of all three compared groups was not statistically significant and it can thus be confirmed that the choice of the GNSS receiver did not affect the accuracy of the PPS signal.

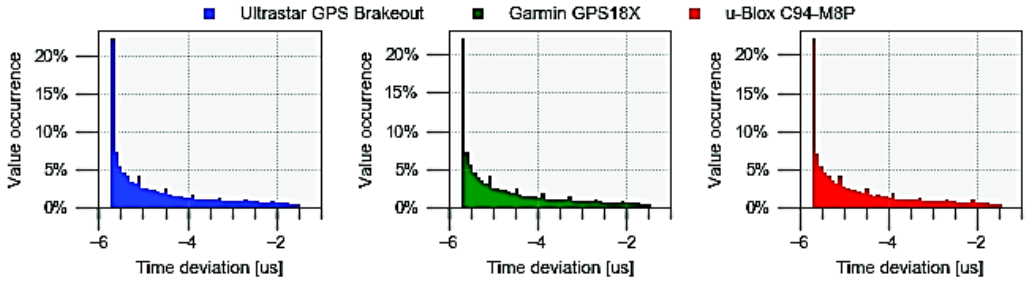

Figure 9. The distribution of time deviations of three PPS signals against the analyzer clock.

Verification of the quantity and correctness of the sent messages according to the arm instantaneous speed

The verification of the correctness of the times and iterators messages was gradually conducted in the intermediate stages during the data processing of sample measurements (Figs 5, 6, 7). The trajectory could be calculated by using the sets of these message pairs. The data was successfully logged at a reasonable density to ensure a future reference trajectory creation. The density as well as the angular distance of points of the reference trajectory of the MRA was defined by the number of received pulses from the encoder between two points and by the length of the arm. In the ideal case of all messages sent and logged, it would get the position information every 3.77 mm.

The SB unit was not fully successful in sending all messages as shown in Fig. 10. This figure indicated a one-dimensional focused view of a trajectory formed from points of the same measurement as in Figs 5, 6 and 7. This section showed twenty milliseconds cut out, when the MRA was moving at a speed causing the most varied values of omitted messages. The vertical axis represents the relative distance of the 'x' coordinates of the MRA from the centre of the axis of rotation of the MRA (to the East). The information about the number of unsent messages between two points was plotted in vertical axis in the graph.

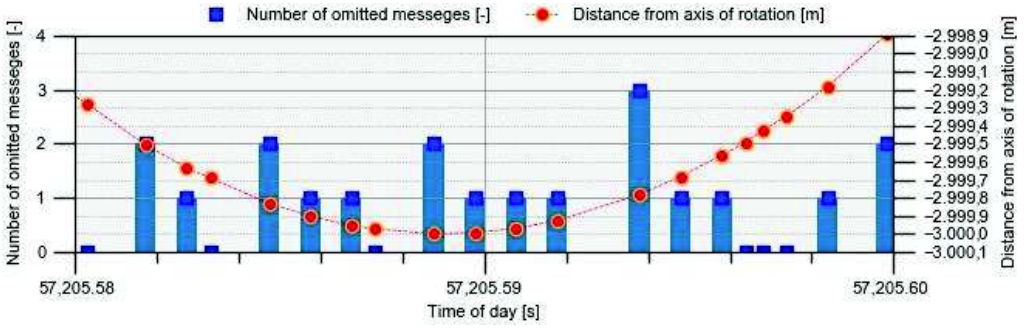

Figure 10. Verification of the quantity of omitted messages.

The verification of the quantity of omitted messages was carried out on a larger dataset of samples in the total number of 271,830 obtained points. Fig. 11 shows the dependency of the number of omitted messages between two measured points (on the vertical axis) on the instantaneous MRA speed (horizontal axis).

During the development phase, it was assumed that the most time-consuming operation for the SB will be the sending of messages on the CAN bus. For these purposes, the algorithm of the SB unit (Fig. 3) was designed to store the location information in the SB memory and to send the correct position of the MRA in the following message. As shown Fig. 10, in case of unsent messages, the position of the points shifts over time and the trajectory doesn't get deformed. To smooth the final reference trajectory, interpolation methods can be used.

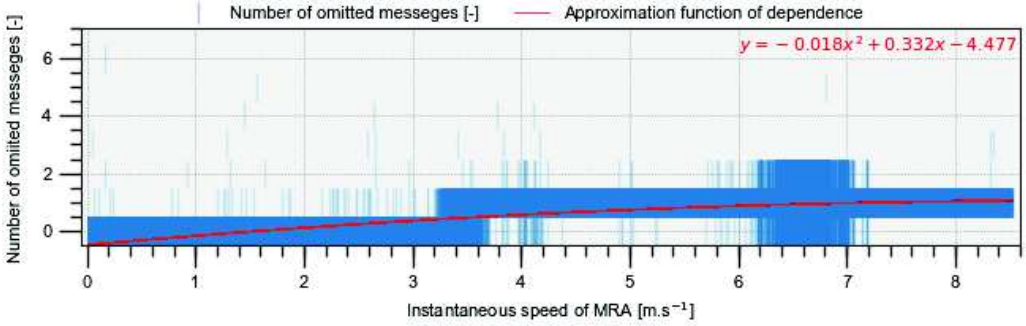

Figure 11. The number of omitted messages between two measured points according to the instantaneous MRA trolley speed.

As shown above, the number of omitted messages had an increasing tendency with the increasing instantaneous angular velocities of the arm. In extreme situations of six omitted messages, the maximum value of the angular distance between two points could be 22.619 mm. Nevertheless, on this large sample of data, such situation occurred only once. Unfortunately, the average value is unsuitable for this evaluating method because it would be burdened by the unevenly distributed number of samples depending on the instantaneous speed.

The real measurement of RTK systems

Since the planned verification of RTK systems implies the introduction of conditions that are difficult to implement and will be object of future authors' studies, only figures expressing the character of the positioning accuracy in the used mode were given. Fig.12 shows the measured RTK points to the Tersus BX-305 receiver whose antenna was placed on the MRA trolley.

Fig. 12, (1) indicates the measurement with no corrections (the RTK receiver worked as a separate GNSS). It shows that the measured points in the short-term periods tracked the circle of the reference trajectory, but from a long-term point of view, the precision of the positioning is unpredictably displaced by the impacts described in the previous authors' study (Kadeřábek et al., 2018). Fig. 12, (2) depicts the measurement of the RTK receiver in 'RTK fix' mode. Better ability to track the reference trajectory can be observed, but the scattering of the deviations of measured points in direction from the

centre of the circle is also evident. This was due to the acceleration and deceleration effects during the measurement process. Fig. 12, (3) finally shows the points measured by the RTK receiver in 'RTK fix' mode together with the activated signal fusion with the sensors of integrated IMU. A very good tracking of the reference trajectory was observed. This was possible probably thanks to a well-designed signal fusion of RTK and integrated IMU in this unit.

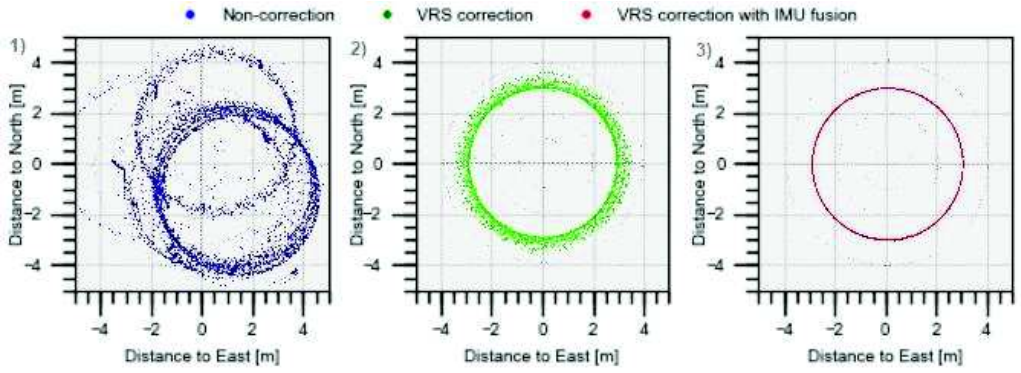

Figure 12. The points measured by Tersus BX-305 occurring in three modes.

As stated above, this rating was not compared with the MRA reference trajectories in given specific values. However, this figurative evaluating indicates that using of the MRA equipment as method to validate RTK receivers described in this paper is useful. The authors assume that further work on evaluations using MRA is needed since it can be beneficial not only in case of RTK receivers, but also for validating other localization systems.

This study has gradually demonstrated that the method using the MRA for the measurement enabled to log the data into the desired form. The data was successfully processed into the trajectory form which could thus serve as a reference system for verifying the dynamic properties of RTK or other localization systems. Previous studies have not so far identified any similar system for a RTK receiver dynamic evaluation. The dynamic evaluation of the RTK positioning receiver properties has already been discussed by some researches (Gan-Mor et al., 2007; Bakker et al., 2011; Carballido et al., 2014; Jilek, 2015; Kabir et al., 2016) where for example tractors or robots were moved in defined or straight paths that were used as reference system. The research of (Boffi, Gilgien & Wieser (2016)) used a downhill coaster track as part of reference unit, but the results focused primarily on velocity estimation, not position estimation. From our literature review it was evident that no study has used such equipment or methods as precise as the method described in this study.

CONCLUSIONS

The goal of this study was to describe the design of the method using the MRA device to verify the accuracy of localization systems on a reference circular trajectory created with absolute coordinates and labeled with absolute time stamps of one day. The design of this method was presented by the description of the mechanical design of the

MRA, a block diagram of electronic interconnection, a process diagram of the data acquisition from MRA sensors and a description of data processing with the aim of calculating the points of the final reference trajectory.

The verification of the present method was performed by several evaluations. First of all, the time accuracy of a selected type of GNNS receiver for the RMA was confirmed based on the examination of the accuracy of its PPS signal for the purposes of the MRA time base. Secondly, the method of data processing was verified together with the demonstration of the influence of the quantity messages sent by the SB according to the instantaneous speed of MRA. Finally, the contribution of this study was presented in the form of graphical representation of obtained results of measurements of the RTK receiver (Tersus BX-305) placed on the trolley on the MRA. In future research, the authors aim to present the findings of the evaluation of the RTK receivers' ability to determine the accurate position during their movement. Moreover, the employment of the method using the MRA device for validations of other localization systems will be verified.

ACKNOWLEDGEMENTS. This study was supported by grant from the Internal Grant Agency of the CULS 'Evaluation of Determined Position Accuracy by RTK Receivers in movement' (No. 31160/1312/3114).

REFERENCES

Bakker, T., Asselt, K.V., Bontsema, J., Muller, J. & Straten, G. 2011. Autonomous navigation using a robot platform in sugarbeet field. *Biosyst. Eng.* **109**(4), 357–368.

Berber, M., Ustun, A. & Yetkin, M. 2012. Comparison of accuracy of GPS techniques. *Journal of the International Measurement Confederation* **45**(7), 1742–1746.

Boffi, G., Gilgien, M. & Wieser, A. 2016. Validation of GNSS-Based High-Precision Velocity Estimation for Outdoor Sports. In: *International Federation of Surveyors FIG Working Week*, Christchurch, New Zealand.

Carballido, J., Perez-Ruiz, M., Emmi, L. & Agüera, J. 2014. Comparison of positional accuracy between rtk and rtx gnss based on the autonomous agricultural vehicles under field conditions. *Applied Engineering in Agriculture* **30**(3), 361–366.

Feng, Y. & Wang, J. 2007. Exploring GNSS RTK performance benefits with GPS and virtual Galileo measurements. In: *Proceedings of Institute of Navigation National Technical Meetings*. San Diego, CA, USA, pp. 22–24.

Feng, Y. & Wang, J. 2008. GPS RTK Performance Characteristics and Analysis. *Journal of Global Positioning Systems* **7**(1), 1–8.

Gao, J., Liu, C., Wang, J., Li, Z. & Meng, X. 2011. A new method for mining deformation monitoring with GPS-RTK. Transactions of Nonferrous Metals Society of China, **21**, 659–664.

Gan-Mor, S., Clark, R.L. & Upchurch, B.L. 2007. Implement lateral position accuracy under RTK-GPS tractor guidance. *Computers and Electronics in Agriculture* **59**(1–2), 31–38.

Garrido, M.S., Giménez, E., de Lacy, M.C. & Gil, A.J. 2011. Surveying at the limits of local RTK networks: Test results from the perspective of high accuracy users. *International Journal of Applied Earth Observation and Geoinformation* **13**(2), 256–264.

Jilek, T. 2015. Autonomous field measurement in outdoor areas using a mobile robot with RTK GNSS. *IFAC-PapersOnLine* **28**, 480–485.

Kadeřábek, J., Shapoval, V., Matějka, P. 2018. Evaluation of the RTK receiver's capability of determination the accurate position. *Agronomy Research* **16**(3), 749–757.

Kabir, M.S.N., Song, M.Z., Sung, N.S., Chung, S.O., Kim, Y.J., Noguchi, N. & Hong, S.J. 2016. Performance comparison of single and multi-GNSS receivers under agricultural fields in Korea. *Engineering in Agriculture, Environment and Food* **9**(1), 27–35.

Mishra, P., Karami, A., Nordon, A., Rutledge, D.N. & Roger, J.M. 2019. Automatic de-noising of close-range hyperspectral images with a wavelength-specific shearlet-based image noise reduction method. *Sensors and Actuators, B: Chemical* **281**, 1034–1044.

Niu, X., Yan, K., Zhang, T., Zhang, Q., Zhang, H. & Liu, J. 2014. Quality evaluation of the pulse per second (PPS) signals from commercial GNSS receivers. *GPS Solutions* **19**(1), 141–150.

Raskaliyev, A., Patel, S. & Sobh, T. 2017. A dynamic model for GPS based attitude determination and testing using a serial robotic manipulator. *Journal of Advanced Research* **8**(4), 333–341.

Zhou, H., Nicholls, C., Kunz, T. & Schwartz, H. 2008. Frequency accuracy & stability dependencies of crystal oscillators. *Carleton University, Systems and Computer Engineering, Technical Report SCE-08-12*, pp. 1–15.