

What Can We Add to Solution Environments to Enable Real-time Teacher Support?

Rein Prank, *rein.prank@ut.ee*

University of Tartu, Institute of Computer Science, Liivi Str 2, 50409 Tartu, Estonia

Abstract

Teachers often observe the work of students in a computer classroom in a rather passive manner. They have no tools for quick acquisition of information about the progress of students' work, the difficulties experienced or the quality of results. This paper proposes a selection of information for recording about the work of students and a selection of tools for giving the teacher both general and detailed information about the state of each student and the group of students as a whole.

Keywords

exercise environments, student solutions, tools for teachers

INTRODUCTION

Many exercise environments check correctness of answers and solution steps of students. They detect mistakes and require correction before the next step/task can be started. Good solution environments are also able to give hints for next step. However, we know that the programs perform only a part of the work of teachers. The programs usually react to one particular mistake by one student. They do not take into account previous mistakes, possible other mistakes in the same step, mistakes and difficulties of other students, etc. The hints for next step are usually derived from a general and formal solution algorithm that does not check for special features of concrete situation. Solution environments leave the most generalized and the most refined analysis to the teacher. At the same time, they usually do not provide the teacher with necessary information about the state of the students' work and the quality of results. In this paper we consider exercise environments for algebra and discuss the information that the environments can provide for teachers to support their work during exercise labs and the following analysis of results.

During exercises in a traditional paper-and-pencil environment the teacher monitors the progress of students and asks them to correct any noticed mistakes. If necessary, he/she can ask them to show solutions from previous pages. However, the situation with available information tends to be different when students work on computers. Prevaingly, any erroneous input and error diagnosis disappear from the screen after entering the next variant of the answer. Most of the programs do not record the entered solutions. This makes it impossible to evaluate solution economy and to identify interesting solutions. Usually it is also impossible to know at what stage the student has broken the solution. In case of most computerised environments the teacher obviously has less information for undertaking necessary pedagogical actions than in the traditional environment. The program does possess necessary fragments of useful information at different stages of the solution process, but the teacher can only see a random selection of these fragments. The remaining part is systematically erased from the memory. Nevertheless, such behaviour is not mandatory for solution environments.

Different programs for making algebraic conversions have very different facilities for recording and reviewing student activities. Many small programs, but also the mathematically sophisticated MathXpert (Beeson, 1998), do not record students' solution steps. On the other hand, Aplusix records the solutions and contains also a 'video tape recorder' that allows to record and replay every action of the student (Sander, Nicaud, Chachoua & Croset, 2005). At the same time, the authors admit that an analysis of this detailed information requires several hours for one hour of one student problem solving session. According to (Libbrecht, Rebholz, Herding, Müller, & Tscheulin, 2012, p. 115), "These logging repository initiatives are researcher-oriented: they bring together quantities of logs in order to formulate hypotheses about the learners' actions. They are not applicable for teachers". AS-SISTment (Feng & Heffernan, 2007) asks from the student only the final answers but also records successive corrections made after one or more hints are given by the system in case the answer was wrong.

The work of the teacher is more efficient if:

1. An exercise environment records analysis-oriented and pre-processed information instead of raw material.
2. There are tools for obtaining necessary views of recorded information.

In this paper we regard the exercise environment T-algebra (Prank, Issakova, Lepp, Tönisson & Vaiksaar, 2007) for basic school algebra exercises as a primary student tool. Corresponding to items 1 and 2 above, we propose a selection of recording/reviewing facilities for the student program and additional group view tools for the teacher.

The next two sections of the paper describe the solution dialog, the available support for students and the reviewing facilities of the student program. After that the main section describes additional teacher tools for working with a group of students: overview tables, error statistics and tools for data mining.

SOLUTION DIALOG AND SUPPORT FOR STUDENTS IN T-ALGEBRA

T-algebra is an interactive learning environment for calculation of the values of integer expressions; operations with fractions; solving of linear equations, inequalities and linear equation systems; operations with polynomials. The solution dialog contains elements of both rule-based and input-based interfaces. We give here only a brief description of the solution step dialog. More detailed descriptions are published in (Prank, Issakova, Lepp, Tönisson & Vaiksaar, 2007).

The student solves expression manipulation tasks step by step. The step dialog of the majority of conversion rules consists of two substeps:

1. Selection of an operation from the menu and marking the operand(s) in expression.
2. Entering the result of the operation.

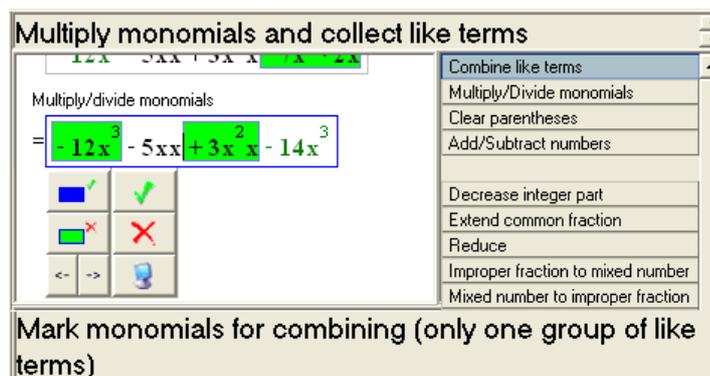


Figure 1: Solution window during the substep 1. For the next step in the solution, the student has selected the operation Combine like terms and has marked two terms for combining.

The student can perform the two actions of the first substep in any order. Figure 1 demonstrates essential parts of the solution window during execution of substep 1.

T-algebra starts to check the operation and marking after the student clicks the green check button on the virtual keyboard. If the operation and operands are acceptable, T-algebra asks the student to enter the result of application of the selected operation to the marked operands (Figure 2).

Combine like terms

$$= \boxed{} - 5xx - 14x^3$$

Figure 2: Input stage after the above selection.

When the expression has been converted to the required form, the student should click the button "Solved".

Support mechanisms

In order to plan his/her pedagogical actions, the teacher should know what support is already provided by the computer. T-algebra provides several types of support for students:

1. Instructions for selection of the operation, marking and entering the result of the step, etc., are given in the lower frame of the solution window.
2. T-algebra contains an automatic solver that implements textbook algorithms for all task types. The student can request help for selecting an operation, marking the operands and entering the result of the step. Availability of each of the help facilities is set up in the task file.
3. Error messages are the most detailed part of the student support. At the first substep, T-algebra checks whether the marked parts are syntactically correct subexpressions and whether they are suitable for selected operation (see Figure 3). At the second substep, T-algebra checks syntactical correctness of entered expression(s) and equality between entered and solver-computed result of the step. For many operations, T-algebra also checks whether the structure of the result corresponds to the performed operation. At the end of solution, the program checks whether the expression (equation, inequality, system) is really in the final form (required for the current task type). In case of mistakes or deficiencies, the program issues an error message, highlights a part of the student's marking/input, and requires correction of the mistake.

Multiply monomials and collect like terms

$-12x^3 - 5xx + 3x^2 - 7x - 2x$

Multiply/Divide monomials

$= -12x^3 - 5xx + 3x^2 - 14x^3$

Combine like terms

Multiply/Divide monomials

Clear parentheses

Add/Subtract numbers

Decrease integer part

ed number

er fraction

ecimal

mon

At least one of the terms is not similar to others

OK

Figure 3: Error message about marked operands.

PRIMARY REVIEWING FACILITIES OF T-ALGEBRA

T-algebra saves the results of the work of one student on one task file in a solution file. The solution file can be located in a folder on the server (and be readable for the teacher) or be hosted by the student and then submitted for checking. The solution file contains student solutions (complete or incomplete) of all tasks, all instances of error and help usage, as well as the times of start and end of work on each task. The student program of T-algebra enables to open the solutions as they appeared on the screen when the student solved the task, and offers five additional views of information recorded in the solution file:

1. Table of errors in tasks (errors are classified in 20 task-independent types);
2. List of errors (with a possibility to open each error situation);
3. Table of help usage in tasks (help requests are classified by the action in solution step);
4. List of help requests (with a possibility to open each situation);
5. Table of general statistics by tasks (numbers of errors and help usage, solution time, etc).

Figure 4 presents a review of the solution to the task shown in Figures 1-3 and a review of one error situation.

$4x^2 \cdot (-3x) - 5xx + 3x^2 \cdot x - 7x^2 \cdot 2x =$

Multiply/Divide monomials

$= -12x^3 - 5xx + 3x^2 \cdot x - 7x^2 \cdot 2x =$

Multiply/Divide monomials

$= -12x^3 - 5xx + 3x^2 \cdot x - 14x^3 =$

Combine like terms

$= -9x^3 - 5xx - 14x^3 =$

Combine like terms

$= -23x^3 - 5xx =$

Multiply/Divide monomials

$= -23x^3 - 5x^2$

Problem nr. Error nr.

Error type **Sign errors**

Selected rule **Combine like terms**

Problem text

Multiply monomials and collect like terms

Problem expression

$4x^2 \cdot (-3x) - 5xx + 3x^2 \cdot x - 7x^2 \cdot 2x$

Error message displayed to the student

Incorrect sign

Description of error

$-12x^3 - 5xx + 3x^2 \cdot x - 14x^3$

$9x^3 - 5xx - 14x^3$

Figure 4: Review of a solution (left) and display of an error situation therein.

ADDITIONAL TOOLS FOR GROUP INFORMATION

The above-described features of the student program of T-algebra provide both general and detailed information about the work of one student. However, the teacher also needs an overview of the work of the entire class or group of students. For example, during and after a session in a computer lab, it would be important to find out how many tasks are solved by each of the students, which tasks caused the most difficulties and errors, which students were not able to master the tasks or the user interface. Having such information, the teacher can provide additional instructions on some tasks or operations, instruct some individual students, etc. When checking homework assignments, the teacher needs quick information about the students who did not solve all the required tasks in their submitted files. All this data can be retrieved from the main program, but it would require opening student files of the whole group one by one, and writing down the collected information. Therefore,

the teacher needs additional tools that enable a top-down approach and present the data of all students together.

At the current stage, the group view software of T-algebra exists in the form of an additional program for creating tables with general data of all students and some scripts for data mining. We start with a description of information presented in tables. Some of our considerations can be relevant to exercise sessions in computer labs, while others pertain to tests or homework.

Tables of results of students' work on each task

The tables are generated by placing together the corresponding rows from individual student tables. The content of the tables can be exported for further processing in spreadsheets or statistical applications.

1. The table of solved/unsolved tasks consists of separate rows for each student and columns for each task. The cells contain symbol '+' if the student solved the task, symbol '-' if the student tried to solve the task but failed, and space if the student did not attempt the task. The row of each student also shows aggregate numbers of tasks that were solved, tasks that were started but not finished. The column of each task shows the corresponding numbers of students.
2. In the table of step counts, the rows of each student show the number of steps, accepted by T-algebra, in each (finished or unfinished) solution and the total number of executed steps. The number of steps is preceded by a minus if the task was attempted but not finished. The first two rows of the table can be used to display the number of steps from two reference files with 'normal' solutions. Numbers of steps in standard solutions of the built-in Solver of T-algebra are an acceptable yardstick in many cases. However, it is possible to add a reference file with teacher's own solutions to fill in the second data row of the table.
3. In the table of solution times, the rows of each student show the time used for each task (finished or unfinished) and the total time used for solutions (in seconds). A minus again indicates an unfinished task. The teacher can use one row of the table to display reference solution times from a solution file.
4. In the table of error counts, the rows of each student show the number of solved tasks, the total number of errors and the number of errors made in each task.

The examples on the following pages show the upper left corners (11 students and 20 tasks) of tables 2-4 of a lab session with 46 students of grade 7. They worked for 45 minutes on a task file containing 30 tasks on multiplication of fractions. The view of solved/unsolved tasks enables to discover the students who have difficulties, who do not work properly in lab session or have submitted uncompleted homework. If many students have attempted, but have not finished a particular task then this can be caused by a user interface issue that has not yet been explained in classroom or by disagreements between completion requirements of the program and the teacher's requirements in paper-and-pencil processes. Such cases can often be solved through additional explanations.

The table of step counts enables to evaluate general solution economy, detect unnecessarily long solutions of particular students, the types of tasks where many students use uneconomic solution routes, the tasks where students have found shorter solutions. The teacher can also discover instances where a specific student has entered a very short solution produced by additional calculations on paper or by using information from the screen of a neighbour. In all interesting cases, it is possible for the teacher to obtain more precise information by opening the student's solutions in the main program.

Table 1: Step counts of 11 students in 20 tasks on multiplication of fractions. #OK denotes the number of solved tasks (of 30). At a certain point in the session, the students were asked to skip some tasks and jump to task 16 (expressions with 3 or more operations).

	#OK	Steps	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Solver	30	230	5	5	5	5	3	5	8	6	4	8	5	7	7	6	7	8	8	5	9	8
Teacher	30	92	3	3	3	3	1	2	8	4	1	6	1	1	1	2	1	4	2	3	5	4
1 Pul.lah	20	102	3	3	3	1	1	1	4	6	2							5	6	5	10	11
2 mit.lah	11	53	3	3	3	3	1	8	6	6	2	-3	5	9	-2							
3 Kik.lah	21	127	3	3	3	3	1	8	6	5	3	6						7	5	4	8	9
4 kuz.lah	12	75	3	3	3	5	1	5	-4									7	7	4	8	11
5 nom.lah	11	47	1	3	1	1	1	5	8	4	4	-5	5	7	-3							
6 Jam.lah	15	108	5	5	5	5	3	9	8	6	2							8	6	5	9	10
7 kar.lah	15	99	5	3	5	5	1	9	8	6	2							8	6	3	9	10
8 Ber.lah	9	24	1	3	3	1	1	2	8	3	1	-2										
9 Sil.lah	13	59	3	3	3	3	1	6	-2	4	2	-4	3	5	9	-2	3	8	0			
10 Pol.lah	16	61	3	3	3	3	1	5	8	1	1	1						5	3	4	6	5
11 zuj.lah	19	49	1	1	1	1	1	2	4	2	2	0						3	3	2	3	5

Table 1 also contains a reference line with the teacher's solutions. The students were expected to reduce (where possible) the numerators and denominators of factors mentally before multiplication. However, solutions of the automated Solver follow the solution algorithms presented in the textbook to the letter and are much longer.

Table 2: Solution times of multiplication of fractions.

Task	file:tund9_korrutamine.yls	Solution time/#OK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Teacher	3193/30	116	67	177	56	49	61	234	88	73	164	25	41	34	89	117	86	72	140	244	69	
1 Pul.lah	2490/20	29	40	21	19	14	56	226	62	53								127	133	280	101	133
2 mit.lah	2247/11	42	67	34	27	21	485	390	76	145-438	98	237-187										
3 Kik.lah	2297/21	38	30	21	21	15	108	84	45	48	292							73	91	213	79	94
4 kuz.lah	2180/12	30	44	93	175	29	167-173											153	265	336	109	133
5 nom.lah	2549/11	39	62	30	27	33	358	662	145	258-616	82	154	-83									
6 Jam.lah	2322/15	76	73	68	44	33	87	164	76	248								164	110	164	104	120
7 kar.lah	2320/15	65	63	106	60	29	129	216	69	108								117	122	182	148	157
8 Ber.lah	2505/9	22	120	32	29	55	59	999	84	45-293												
9 Sil.lah	2183/13	115	48	38	22	22	173-353	61	143-231	46	59	149-281	32	241-169								
10 Pol.lah	2418/16	203	155	38	24	24	125	340	16	33	34							169	122	281	128	215
11 zuj.lah	2274/19	37	54	20	16	14	63	146	41	46	-11							85	109	195	93	98

The table of solution times enables to discover the tasks and subtopics that have proven to be difficult for some or even for all students. This analysis can be further complemented by juxtaposing it with the table of error counts.

The tasks 6, 7 and 10 have been difficult for many students (see also Table 3 with error counts). Opening the solutions and error descriptions of task 7 we see that most students have converted mixed numbers successfully to improper fractions (the first two steps in Figure 5). Many of them then multiplied the fractions without previously reducing by 7. This caused many calculation errors and loss of time. Some students wanted to reduce but had troubles with the solution interface of T-algebra. Naturally, T-algebra does not allow reducing numbers from different fractions (error situation in Figure 5). The menu also does not contain any special conversion rule like 'Present the product as one fraction'. However, there is a loophole: the rule Multiply/divide numbers allows, in essence, to omit the multiplication of fractions (third step in left side of Figure 5) and so prepare the fractions for reducing. Some students found the way but some others even abandoned the task. It would be quite easy to give an appropriate hint if the teacher could recognise the situation in time.

Selected rule **Reduce**

Problem text

Multiply fractions

Problem expression

$$2\frac{5}{7} \cdot 3\frac{1}{9}$$

Error message displayed to I

Marked operands cannot be multiplied

Description of error

$$\frac{19}{7} \cdot \frac{28}{9}$$

Figure 5: Three steps from a solution of task 7 and an error description.

Table 3: Numbers of diagnosed errors in tasks. The first row contains the numbers of errors of the whole group (46 students).

Task file:tund9_korrutamine.yls		Errors: #Err/#OK																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Group:		31	46	24	18	13	154	234	45	34	104	23	29	21	20	9	35	39	62	38	39
1 Pul.lah	55/20	0	2	0	0	0	0	7	0	1							2	4	4	3	4
2 mit.lah	35/11	0	1	1	0	1	14	6	2	2	-3	0	2	-3							
3 Kik.lah	19/21	0	0	0	0	0	1	0	0	0	1						1	1	4	0	2
4 kuz.lah	12/12	0	0	1	3	1	1	-1									0	0	1	0	1
5 nom.lah	33/11	0	0	0	0	0	9	5	3	4	-8	2	2								
6 Jam.lah	11/15	0	0	0	0	0	0	0	0	2							0	2	1	0	0
7 kar.lah	25/15	0	0	3	0	0	4	3	0	1							1	0	1	0	1
8 Ber.lah	17/9	0	2	0	0	2	0	10	1	0	-2										
9 Sil.lah	37/13	4	1	1	0	0	6	-11	0	1	-1	1	0	1	-4	0	5	-1			
10 Pol.lah	16/16	1	1	0	0	1	0	5	0	0	0						0	1	1	1	1
11 zuj.lah	29/19	1	0	0	0	0	1	1	0	0							0	2	1	1	0

The obvious benefit of this table is that it enables to identify the tasks that have caused more errors. The teacher can then discuss the underlying mathematical or user interface issues with the students. The table of errors also enables to find tasks where some specific students have had problems.

Some students (like student 1 in the table) solve standard mathematical tasks quite quickly, but their method of trial and error produces a large number of errors. The teacher should pay attention to students whose count of errors is larger than expected.

Error Types in Tasks

Our software allows producing two overview tables of errors made in tasks. The rows display 20 types of errors in the first table and particular error messages in the second table. The error messages in the second table are ordered by frequency. The view of errors in the second table is more representative of individual topics, but this table can be much longer and its lower part usually contains many mistakes caused by occasional misclicks. The columns correspond to tasks and each cell contains the aggregate number of errors (of given category/with given error message) of the entire study group. The rows also show the total number of errors (of respective category/with given message) and the columns show the total number of errors in the particular task.

Table 4: Seven most frequent error messages of 46 students.

```

Task file:tund9_korrutamine.yls
Tasks:      1..30  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
-----
1 Calculation error
      441  9 20  2  4  4 55 49 14 13 40  6 15  8  8  3 11 13 34  9 16
2 Marked operands cannot be reduced
      119  4  7  4  4  - 11 34  6  -  9  3  -  5  1  -  4  3  4  4  -
3 All operands should belong to one product/division
      104  6  3  4  2  -  8 29  3  1  5  1  3  2  -  1  9  5  1  4  4
4 Reduce fractions in final answer
      98  4  7 11  3  -  8  2  4  - 14  8  8  3  1  3  3  2  -  6 11
5 Convert improper fractions to mixed numbers in final answer
      63  -  -  -  -  - 24 21  -  -  6  -  -  -  3  -  -  -  6  3  -
6 Some of marked operands cannot be reduced
      52  2  -  -  1  -  1 23  -  4  7  -  -  -  3  -  -  1  6  -  -
7 Missing sign
      51  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  1  -  2  2  -

```

The tables enable to see the most frequent types of errors, to locate their occurrence in tasks and task types, and to analyse dynamics of errors. Some errors reflect the basic skill level of students (for example, calculation errors or errors with conversions that have been learnt earlier) and, in this case, quick improvements are not possible. Other errors are related to recently learnt material, which is the focus of training in the current session. A part of those errors can be associated with appearance of new task types or new elements in the solution, and they become less frequent or disappear in the process of learning to follow the tasks. The same is true for errors that are caused by the user interface of the program. The teacher can help the students if this is not the case and the error level remains high, despite learning.

Data mining

There are several kinds of situations/expressions that inform the teacher about the need of common or individual explanations. Most critical troubles are probably deadlock situations where the student is not able to perform the next step and his/her attempts result only in error messages. Our deadlock detection script scans all solution files and finds consequent error situations with the same starting expression. The program outputs the name of the solution file and detected situations with the current expression and error messages (numbered as in the error view of the main program). For example:

```

{19/7}*{28/9}
22. Marked operands cannot be reduced
23. Marked operands cannot be reduced

```

In task 7, analysed above, the longest deadlock group contained 8 error messages. Often the repeating error messages are formally calculation errors. In case of fractions wrong results can be caused by erroneous understanding of operations, and explanations of the teacher can help the students.

In some cases the teacher wants to know whether the students have followed his/her recommendations or not. The task file for the above session contains several tasks where the initial expression includes reducible fractions. For example, task 10 requires calculation of product. Many students did not reduce the fractional part of the first factor. They then made calculations with bigger numbers and arrived in situation where they were not able to convert $\frac{510}{27}$ to mixed number or to reduce the fraction. Our search tool enables to scan all solution files and to find what files contain occurrences of a given expression (only in solutions of a particular task if desired); for example, occurrences of $5\frac{2}{3}$ in task 10. The same is possible for occurrences of a given expression in error descriptions

CONCLUSION

In this paper we described software solutions for providing the teacher with information required for real-time pedagogical decisions during computerised exercise labs and for later evaluation of the quality of student solutions in algebra.

Our approach relies on a solution environment where the student solves tasks step by step, the program diagnoses the mistakes and requires correction of them. The program records constructed solutions, error situations, help requests and solution times. This enables the teacher to receive both general and detailed information about the work of each student and about the whole group of students.

The ideas behind our software are not particularly specific to algebra. The necessary conditions for using our approach are step-by-step solutions and nontrivial error diagnostics.

Acknowledgements

Current research is supported by Targeted Financing grant SF0180008s12 of the Estonian Ministry of Education.

REFERENCES

- Beeson, M. (1998). Design Principles of Mathpert: *Software to support education in algebra and calculus*. In Kajler, N. (Ed.), *Computer-Human Interaction in Symbolic Computation* (pp. 89-115), Springer.
- Feng, M. & Heffernan, N. T. (2007). Towards Live Informing and Automatic Analyzing of Student Learning: Reporting in ASSISTment System. *Journal of Interactive Learning Research*, 18, 207-230.
- Libbrecht, P., Rebholz, S., Herding, D., Müller, W., & Tscheulin, F. (2012). *Understanding the Learners' Actions when Using Mathematics Learning Tools*. In Jeuring, J. et al. (Eds.): *CICM 2012, LNAI, 7362* (pp. 111–126), Springer.
- Prank, R., Issakova, M., Lepp, D., Tõnisson, E. & Vaiksaar, V. (2007). *Integrating Rule-based and Input-based Approaches for Better Error Diagnosis in Expression Manipulation Tasks*. In Li, S., Wang, D., Zhang, J. (Ed.). *Symbolic Computation and Education*. (pp. 174-191) Singapore: World Scientific.
- Sander, M., Nicaud, J.-F., Chachoua, H. & Croset, M.-C. (2005). *From usage analysis to automatic diagnosis: The case of the learning of algebra*. In *Proceedings of 12nd International Conference on Artificial Intelligence in Education (AIED 2005)*, (pp. 45-52), IOS Press.

Biography



Rein Prank is Associate Professor of the Institute of Computer Science of the University of Tartu. Graduated from the Moscow State University in 1973. PhD in Mathematical Logic (1982, Moscow State University). Teaching subjects: Mathematical Logic, Discrete Mathematics, Computer Aided Learning Software.

Copyright

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/>