Diabetes Automata: Software Engine for Blood Glucose Level Simulation

Aleksandr Agafonov[a], Eirik Årsand[a,b], Alexandra Makhlysheva[b], Håvard Blixgård[b], Meghan Bradway[b], Gunnar Hartvigsen[a,b]

[a] *Department of Computer Science, University of Tromsø - The Arctic University of Norway, Norway*
[b] *Norwegian Centre for Integrated Care and Telemedicine, University Hospital of North Norway*

Abstract

For individuals with diabetes to live a healthy life, they must balance carbohydrate intake, insulin injections, physical activity, and monitoring of blood glucose levels. For adolescents with diabetes, this can be challenging – measuring blood glucose and injecting insulin often feels awkward, especially in public areas. In the long run, the result of not managing the factors affecting their diabetes can cause serious consequences. In order to motivate this particular group with diabetes to maintain recommended levels of blood glucose, serious games can be used. The Diabetes Automata project is a part of the research being done on serious games for diabetes at the Norwegian Centre for Integrated Care and Telemedicine. In the project, we have developed a prototype version of a software engine on which diabetes-related serious games, simulators and other tools for diabetes management on various platforms can be based. The Diabetes Automata calculates blood glucose levels based upon relevant patient-gathered data such as insulin, carbohydrates, physical activity, and the user's biometry. The prototype is not yet evaluated.

Keywords:

Blood Glucose Simulation, Diabetes, Mobile Application, Software engine

Introduction

Diabetes mellitus is a group of metabolic diseases characterized by high blood glucose (BG) levels resulting from defects in insulin secretion, its action, or both. It is a chronic disease that occurs either when the pancreas does not produce enough insulin, or when the body cannot effectively use the insulin it produces. WHO estimates that 347 million people worldwide have diabetes [1]. Type 1 diabetes (T1D) is characterized by absolute deficiency of insulin production, daily administration of insulin, and occurs in about 10% of the cases [1]. The cause of type 1 diabetes is not known and it is not preventable, according to current knowledge about the disease [1].

Good management of one's BG levels makes living with diabetes much easier. Most people with T1D use a blood glucose meter for BG monitoring and insulin pen or pump for insulin injections. Through frequent measurements and injections of different types of insulin, people with diabetes can generally maintain a BG level within the recommended range. For adolescents with T1D, managing this important element can be a challenge. Measuring BG and taking insulin can be perceived by adolescents as stigmatising activities. Sometimes patients forget to administer insulin injections or underestimate the effect of the amount of carbohydrates they have eaten.

However, neglecting the disease leads to serious long-term consequences. Therefore, adolescents must be made aware of the long-term consequences of poor blood glucose management and be motivated to adequately manage their diabetes– for example, using their own media channels. One such channel for achieving this is serious games.

Diabetes Automata is a master's project under development that will be finished during the summer 2015. In this project, we have developed a software engine prototype that can be used in diabetes-related serious games, simulators and other tools on various platforms. The engine calculates and provides the blood glucose level based upon relevant patient-gathered background data such as insulin, carbohydrates, physical activity, along with the user's biometry such as age, gender, height and weight.

Materials and Methods

The design of the Diabetes Automata has been divided into two parts: the engine itself, and a demonstrator application (app) that is used to test the engine's functionality and illustrate its potential use. The engine is developed in Java to make it possible for external applications to use it through the engine's API (Application Programming Interface).

The Diabetes Automata offers the opportunity to run blood glucose simulation at various speeds that can be set through the demonstrator application's user interface: from 1 second per second (real-time) to 99 minutes per second. The demonstrator app is an Android-based simulator with a simple design, which displays various types of output information in a form of text, numbers, and graphs. The prototype can run a customized simulation with the opportunity to save the current state of the simulation and then load and continue it later. In addition, simulations can be based on records from the Diabetes Diary, a mobile self-management application for people with diabetes developed by the Norwegian Centre for Integrated Care and Telemedicine (NST) [2-4]. See Figure 8.

The Diabetes Automata has a modular architecture, which provides certain advantages. For example, it is easier for the engine developers to add, change, and edit code to play with different formulas and equations, and to run the modules in parallel. The architecture is shown in Figure 1.

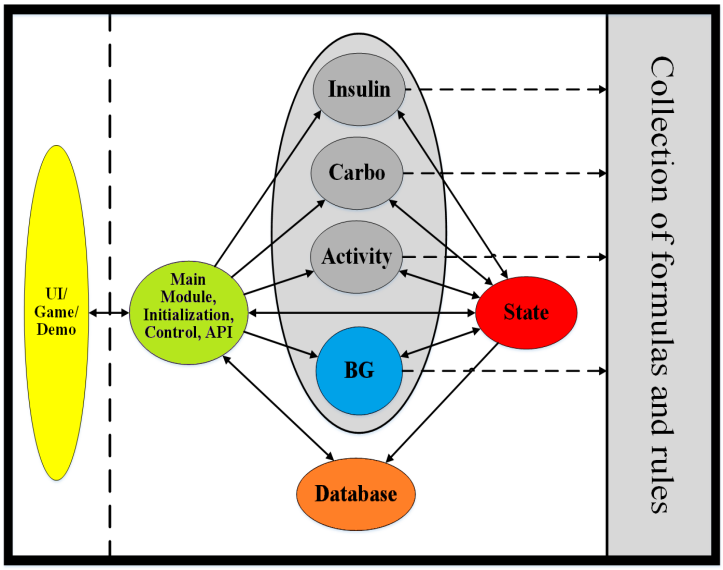

Figure 1 - Diabetes Automata Engine Architecture

The engine contains several modules that are described below.

Carbohydrates Module

First, the engine calculates the user's glucose sensitivity, which is the number of grams of glucose (or carbohydrate with glycaemic index (GI) = 100) that increase the blood glucose level by 1 mmol/L. The calculation of Glucose Sensitivity is based upon Table 1 and the set of functions represented in Figure 2.

Table 1 - BG rise by 1 g glucose depending on body weight. (Source [5])

Body weight (kg)	1 gram glucose will raise BG by (mmol/L):
16	1.11
32	0.56
48	0.39
64	0.28
80	0.22
95	0.18
111	0.17
128	0.14
143	0.12

Glucose Sensitivity (g/mmol) is calculated as follows (see Figure 2). The engine calculates the number of mmol increased by 1g of glucose (y) for the user's weight (x), and then divides 1 by the result.

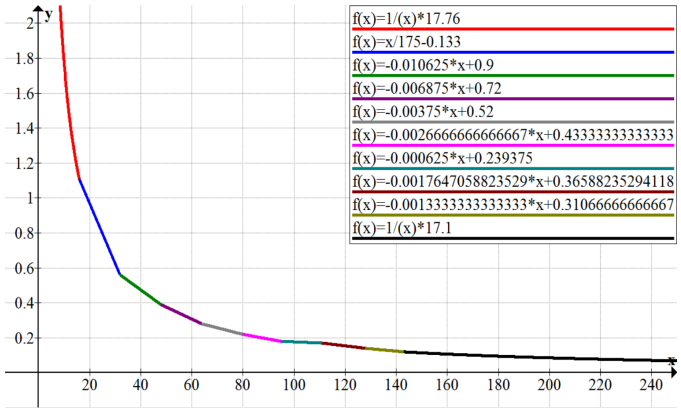

Figure 2 - Glucose Sensitivity in mmol/L per gram of glucose, where x is weight and y is a number of mmol increased by 1 gram of glucose. Figure is created by the authors and based on the data from Table 1 (the data is used as (x,y)-coordinates joined to each other).

Further the duration of action is calculated. The duration depends on GI (see Figure 4) of the engine record and is calculated by the function represented in Figure 3.

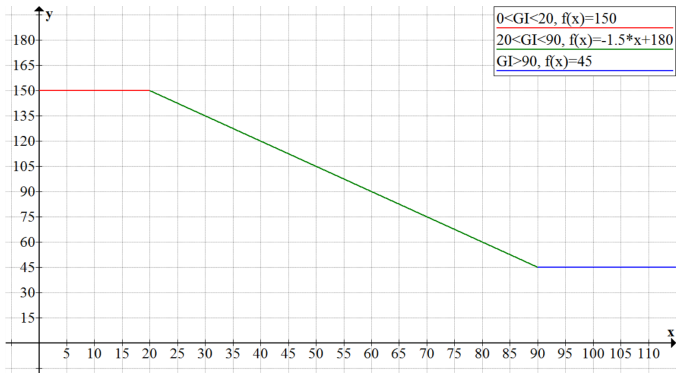

Figure 3 - Calculation of the duration of action of a carbohy-drate simulation record. X-axis represents GI, Y-axis repre-sents minutes.

Figure 4 shows the difference between the duration and the effect of high GI and low GI, for a non-diabetic person.

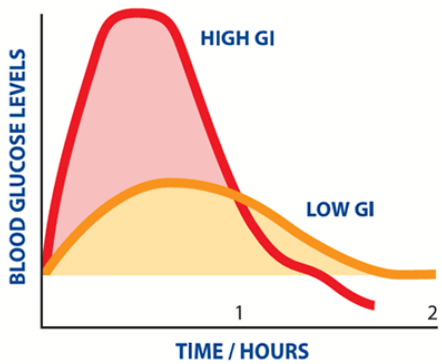

Figure 4 - Duration and action of high GI and low GI, gener-ally. (Source [6])

Next step of the algorithm is to calculate the influence of work done for the current moment in time, in percent. The result

depends on the duration, the example curves for 45, 60, 90 and 120 minutes are represented in Figure 5.

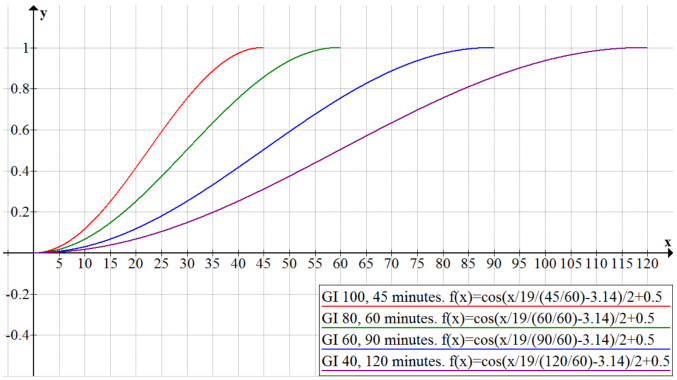

Figure 5 - Percentage BG increase from high GI food over time, where x is time in minutes and y is the effect. 1 on Y-axis represents 100%.

The last step done by the Carbohydrates Module is to calculate the rise in blood glucose in the current moment in time. The calculation is as follows.

```
BG Rise (mmol/L) =
Number of carbohydrates in the record (g) *
GI / 100 (%/100) *
Work done for the moment in time (%/100) /
Glucose sensitivity (g per 1 mmol/L)
```

Before the module finishes its execution, it calculates the additional module values, such as the total module current influence on blood glucose level for the current moment in time, the total module influence for the whole activity time, the relation those two values in percent, the number of currently active records, the remaining module action time, and the history of BG influence value for future purposes. All this is stored to the simulation state. If the action time of the current record is over, it is marked as deleted.

Insulin Module

First, durations of action of an insulin simulation record are defined, which varies between different types of insulin.

The next step is to define the peaking value (1) in mg/min/kg for the defined number of IU/kg (2). Authors found curves representing glucose utilization rate for Humalog, Novolog, Regular, NPH, Levemir, Lantus (see Figure 6) from information sources, and took the maximal glucose utilization rate values for each mentioned insulin type. That was 6.5 mg/min/kg for 0.2 IU/kg for Humalog, 6.8 mg/min/kg for 0.2 IU/kg for Novolog, 5.(6) mg/min/kg for 0.2 IU/kg for Regular, 3.4 mg/min/kg for 0.3 IU/kg for NPH, 3.0 mg/min/kg for 0.8 IU/kg for Levemir, and 0.95 mg/min/kg for 0.3 IU/kg for Lantus.

These values were used to calculate respective coefficients, which will be used in the next step.

```
Coeff (mg/kg/min) = number of IU in the
record * (1) / ((2) * weight)
```

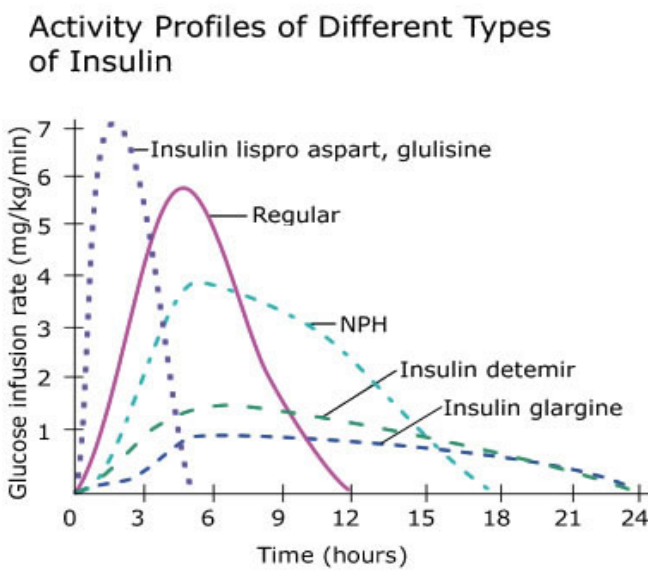

Figure 6 - Glucose utilization rate for different types (examples) of insulin. Rapid-acting insulin is represented by Lispro and Aspart. Short-acting insulin is represented by Regular. Intermediate-acting insulin is represented by NPH. Long-acting insulin is represented by Detemir and Glargine. Source [7]

The next step is to find the current influence and the full influence of the record, by calculating the area under the action curve for particular type of insulin. This is implemented as a loop that iterates through the whole duration line: from zero, when the record was made, until the defined duration, and outputs a sum in mg/kg. The influence is measured in mg/kg/min, so each iteration step is defined as 1 minute.

Each iteration is as follows.

```
Res += Coeff (mg/kg/min) * Influence by
the moment in time (%/100)
```

The functions used for calculating the influence over time build the action curves, which are estimated versions of the curves described in the previous step (see Figure 6 as general example). An example of such estimated curve for the insulin type Humalog is shown in Figure 7.

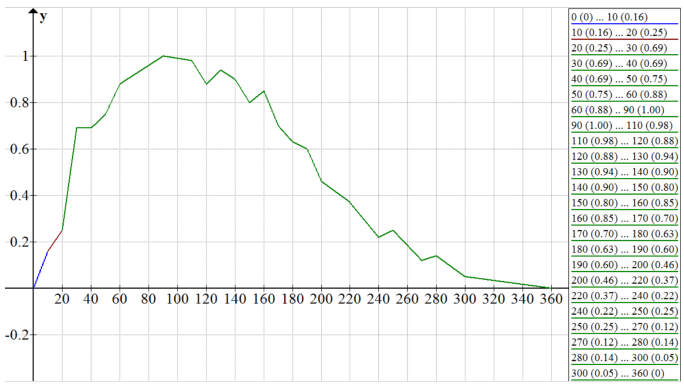

Figure 7 - Estimated action curve for Humalog. X-axis repre-sents minutes; Y-axis represents action from 0 to 1.

The result of this loop is the total influence during the given time in mg/kg. Current influence is calculated in the same way but with one difference – the loop runs not until the end of the duration, but until current moment.

The last step is to convert the results into mmol/L. We know that 1 mole of glucose is 180.15588 g of glucose, or 1 mmol of glucose is 180.15588 mg of glucose. In addition, both insulin and glucose are distributed in the extracellular space in the body. Extracellular space consists of interstitial space (~16% of weight) and blood plasma (~55% of blood volume). So the following conversion was implemented:

```
# (mmol/L) = # (mg/kg) * weight (kg) *
(0.16 + (0.55 * blood volume / weight)) /
     (180.15588 * blood volume (L))
```

Now, we have the current influence and the total influence that will be made by an engine insulin record. Before the module finishes its execution, the same algorithm as in the end of Carbohydrates Module runs. All calculated values are written to the state, and the records which action is finished are marked as deleted.

Activity Module

Aerobic and anaerobic activities have different effects on BG level. According to Bacchi and coauthors [14], aerobic activity causes 30% increase in Insulin Sensitivity because muscles absorb more glucose (which causes a drop in BG) [9] [10] [11]. This increase lasts for the next 24-72 hours after the exercise, depending on the duration of the exercise. During the first 1-3 hours, anaerobic (resistance) activity increases BG by 2 to 4 mmol/L [8], and sometimes even more [13], which is caused by adrenalin [12]. After this, the 15% increase of Insulin Sensitivity [14] takes effect and lasts about 14-20 hours [8]. This information is the basis of the Activity module algorithm.

Blood Glucose module

This module calculates the resulting blood glucose from the Insulin, Carbohydrate and Activity modules.

Additionally, every time the module is called, it subtracts a small value representing the constant glucose consumption by brain (about 100g per day, or about 4g per hour [15]).

Next steps will be to implement at least two other important elements that the Engine should have: Hypoglycemia unawareness option, and Dawn Phenomenon simulation (the rise in blood glucose during the night caused by the hormones). If they are implemented in the future, they will be developed in Blood Glucose module.

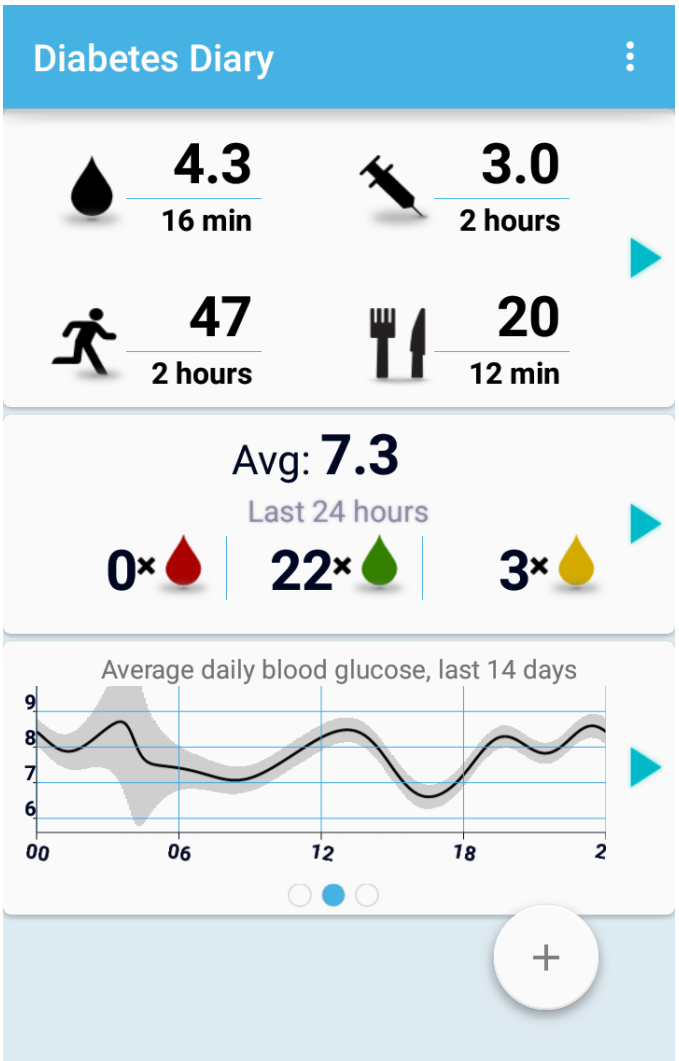

Figure 8 - Diabetes Diary, a mobile self-management applica-tion for people with diabetes by NST.

Main Module

The main Diabetes Automata module initializes the engine and runs the simulation loop repeatedly where it calls Insulin, Carbohydrates, Activity and Blood Glucose modules with the user-specified frequency. For example, it is also possible to run one simulation-hour per minute. Additionally, it provides the API (Application Programming Interface) for external applications to use the Engine code in e.g. serious games or simulation-based self-management applications.

Results

The current version of the Diabetes Automata is a functioning prototype, presented in Figure 9 - Figure 16. Figure 9 presents the page for engine customization (settings), which includes age, gender, height, weight, insulin types used, preferred language, and other parameters.

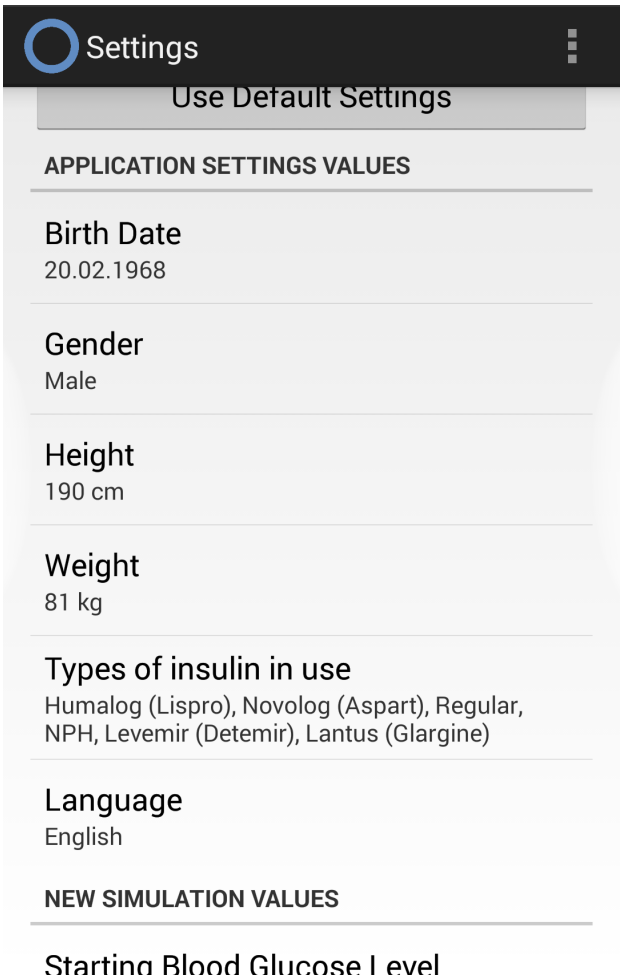

Figure 9 - Settings screen

The prototype provides three simulation modes (see Figure 10). The first mode is the usual simulation where the user inputs new records into the Engine by him/herself. The second mode is the simulation that uses records from Diabetes Diary database by fetching the database records and inputting them into the engine automatically, during the simulation time. This mode is currently personalized for one of the test persons, and has the following default settings: the Engine takes records from Diabetes Diary, use of insulin, long-lasting insulin is Lantus, carbohydrates have medium GI, and physical activity is aerobic. The third mode is a real-time simulation paired with Diabetes Diary. It starts from scratch without the opportunity to change the simulation speed or pause. From this, every time user makes a new record in Diabetes Diary, the record is automatically added to the running simulation.

Figure 11 and Figure 12 show the main screen of Diabetes Automata with started simulation, offering all information necessary to understand the simulation. We can see that the simulation is set to 10 Min / 1 Sec. This means that for each second the result for a ten minute interval is presented. The duration field presents the progress in simulation-time. Time 01:39 (see graph) is the current simulation time between 00:00 and 23:59.

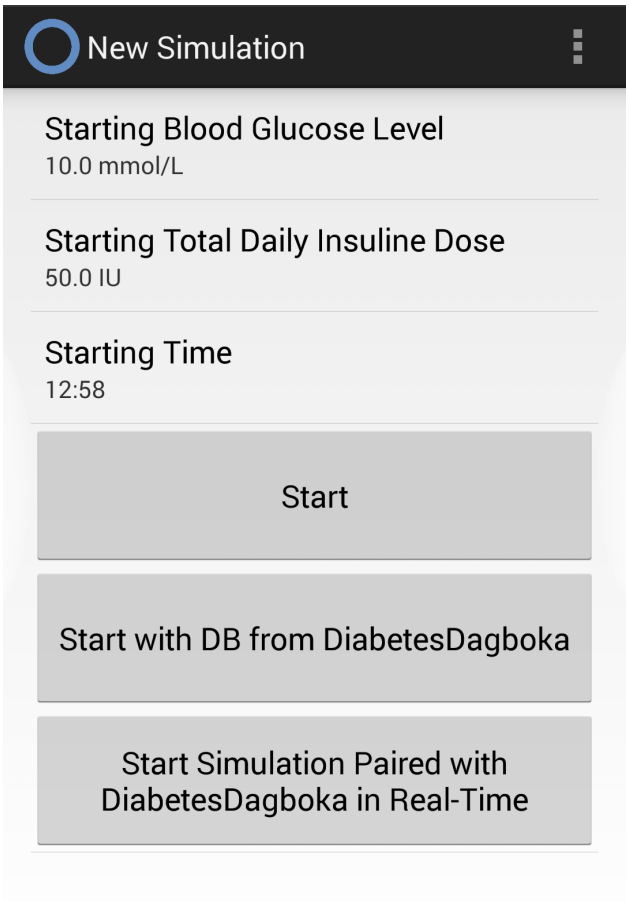

Figure 10 - New Simulation screen

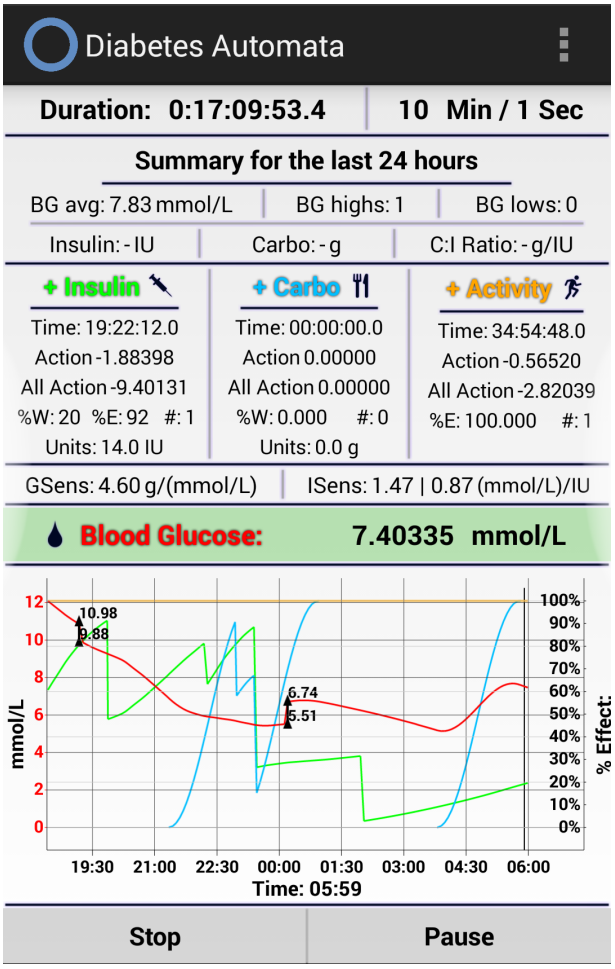

Figure 11 - Simulation. Main Screen

The meaning of presented Insulin, Carbohydrates and Activity values in three columns on the main screen follows these calculations. The first value in each column is the time until that the longest effect of simulation records will last. The second value shows current effect of these simulation records on BG level. The third value is the expected maximal effect of these simulation records on BG. The set of values after, displays the work (%W) and the current effect (%E) of these simulation records on BG level in percentage, and the number (#) of currently active simulation records. The last value in Insulin, Carbohydrates modules is the number of currently active module units.

Blood glucose is represented in mmol/L, and insulin, carbohydrates, and activity curves are represented in percentage of influence. The small black triangles with values (see graph) mean that a user manually calibrated the BG level.

Several functionalities are available for the user on the main screen. One can make a new simulation record by pressing on Insulin, Carbohydrates, and Activity, manually calibrate BG level by pressing Blood Glucose, change the simulation speed by pressing on the time field above Summary, change the graph mode by pressing on the graph, and stop or pause the simulation by pressing the buttons below the graph.

The user can also save the entire current simulation, load it later and continue. This can be done via the options menu in the upper right corner of the screen (see Figure 11). This option is available only for the first simulation mode.

The graph can be set to display the last 1, 2.5, 6.5, 12, 24, 48 simulation-hours, or set to the automatic drawing mode which sets the zoom to one of these values depending on the simulation speed. The graph also has different curve modes. First mode (see Figure 11) represents blood glucose levels, together with action progress (in %) for Insulin and Carbohydrates, and the action strength for Activity. The second mode (see Figure 12) differs with the insulin representation. It is represented as the action strength over time (in %). The third mode is just for blood glucose levels. The user may adjust the graph size by selecting a size menu, presented after holding a long-tap during the simulation (see Figure 13).

When the Pause button is pressed, the graph on the screen contains all information from the beginning of the simulation, which can then be zoomed in or out and scrolled within. The default pause graph shows the last 24 hours. The "Summary" shows the number of blood glucose lows, highs and average level, as well as the total amount of grams of carbohydrates, IU of insulin, and the relation between the latter two, for the last 24 hours.

One of the effects that can be studied with the help of Diabetes Automata is the different types of insulins' influences on the blood glucose level (Figure 14). The challenge for this project was to handle the complexity and variety of insulin types.

The Diabetes Automata aims to emulate the body's metabolism. The key input parameters for the engine are insulin, blood glucose, carbohydrates and physical activity. Each of these parameters has a set of alternatives, as illustrated in Figure 14 - Figure 16.

The final version is expected to be completed during the summer 2015.

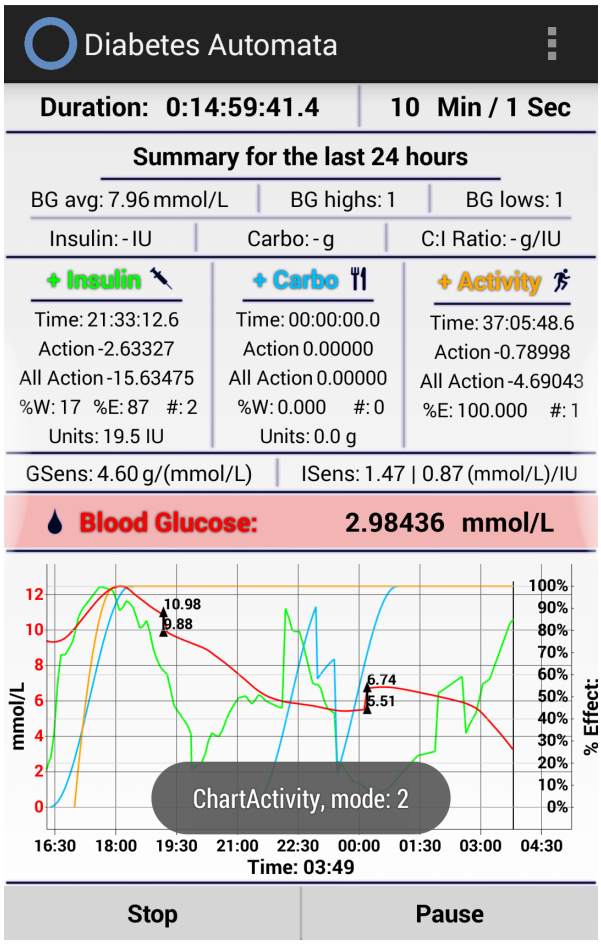

Figure 12 - Main Screen, Simulation

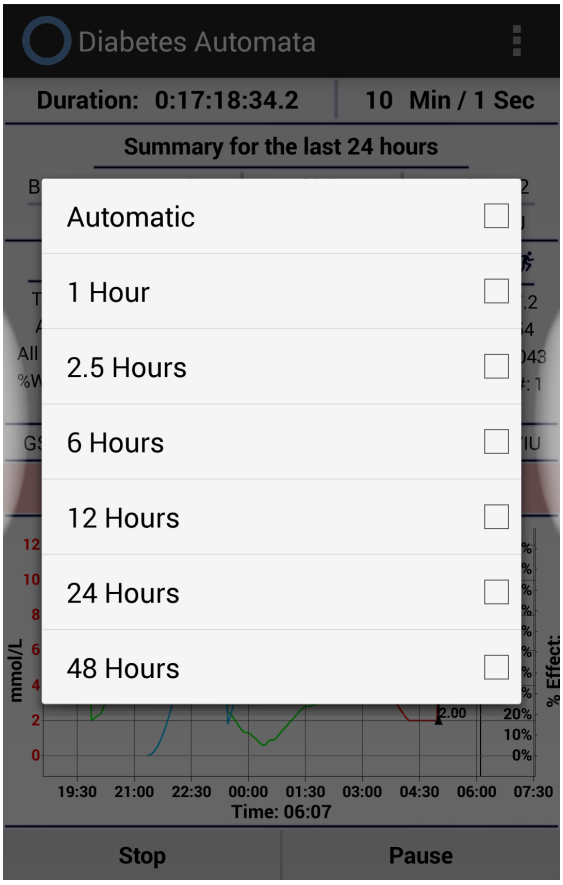

Figure 13 - Choosing the graph zoom modes

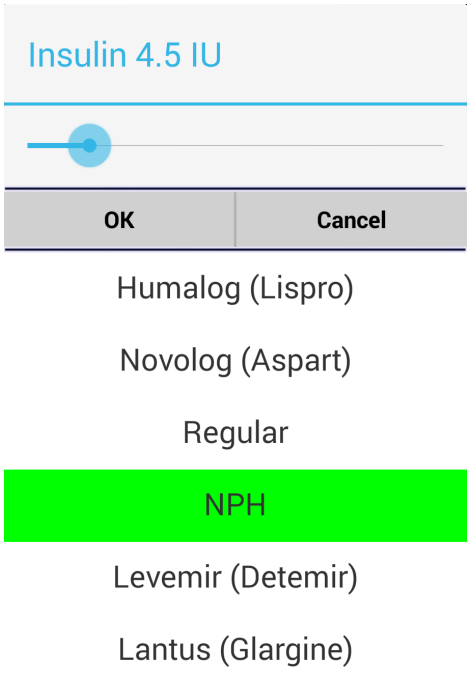

Figure 14 - Make a new insulin registration during simulation

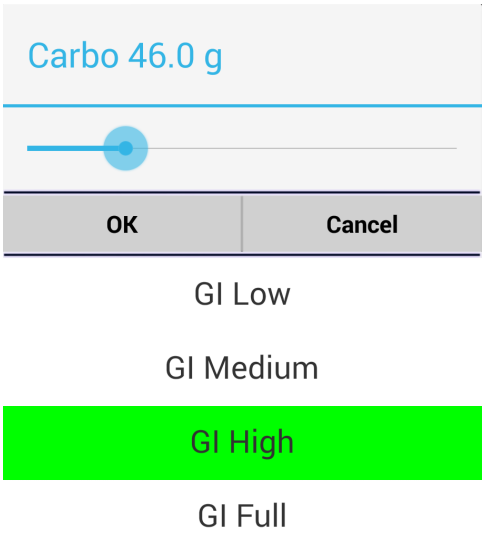

Figure 15 - Make a new carbohydrate registration during sim-ulation

Conclusion

We have developed Diabetes Automata, a software engine that offers the opportunity to run blood glucose simulation at user-defined time-speed.

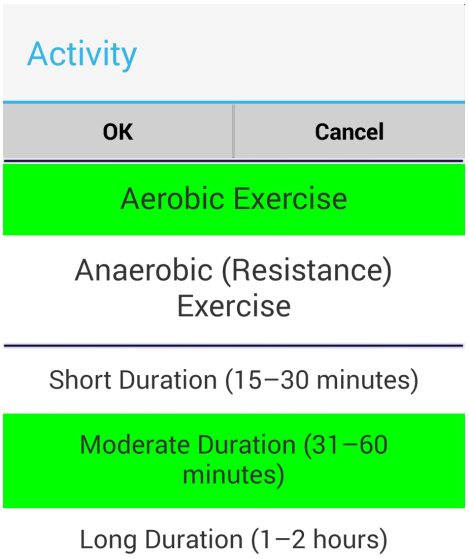

Figure 16 - Make a new activity registration during simulation

The main challenge with the simulator is to make it as realistic as possible. The Diabetes Automata aims to emulate the body's metabolism. We expect that our simulator can be used as part of serious games and contribute through this to a better understanding of the interplay between insulin, carbohydrates, physical activity, and blood glucose levels. This could make games closer to reality and, therefore, more motivational.

The system is not yet evaluated or tested, and is currently under development.

Acknowledgments

This project is partly supported by the Research Council of Norwegian, grant no. 174934.

References

[1] World Health Organization. *Diabetes, Fact sheet N°312*. [Accessed 2015 26 April]; Available from: http://www.who.int/mediacentre/factsheets/fs312/en/

[2] Nasjonalt senter for samhandling og telemedisin. *Diabetesdagboka*. [Accessed 2015 26 April]; Available from: http://telemed.custompublish.com/better-quality-of-life-for-patients-with-diabetes.5208977-287081.html

[3] Årsand, E., Tatara, N., Østengen, G., Hartvigsen, G. Mobile-Phone-based Self-Management Tools for Type 2 Diabetes – The Few Touch Application. *Journal of Diabetes Science and Technology*, March 2010, 4(2), 328-336.

[4] Årsand, E., Frøisland, D.H., Skrøvseth, S.O., Chomutare, T., Tatara, N., Hartvigsen, G., Tufano, J.T. Mobile Health Applications to Assist Patients with Diabetes: Lessons Learned and Design Implications. *Journal of Diabetes Science and Technology*, September 2012, 6(5), 1197–1206.

[5] Bernstein, Richard K., *Dr. Bernstein's diabetes solution : the complete guide to achieving normal blood sugars*, 2007, 325

[6] GlycemicIndex.com, *About Glycemic Index. [Accessed 2015 29 April]; Available from: http://www.glycemicindex.com/about.php*

[7] DiabetesEducationOnline, *Types of Insulin [Accessed 2015 29 April]; Available from: http://dtc.ucsf.edu/types-of-diabetes/type2/treatment-of-type-2-diabetes/medications-and-therapies/type-2-insulin-rx/types-of-insulin/*

[8] Adams, O. Peter, The impact of brief high-intensity exercise on blood glucose levels. *Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy*, 2013, 6, 113-122

[9] AmericanDiabetesAssociation, *Factors Affecting Blood Glucose [Accessed 2015 29 April]; Available from: http://www.diabetes.org/living-with-diabetes/treatment-and-care/blood-glucose-control/factors-affecting-blood-glucose.html*

[10] Diabetes.co.uk, *What Affects Blood Sugar Levels [Accessed 2015 29 April]; Available from: http://www.diabetes.co.uk/blood-glucose/what-affects-blood-glucose-levels.html*

[11] AmericanDiab, etesAssociation, *Blood Glucose Control and Exercise [Accessed 2015 29 April]; Available from: http://www.diabetes.org/food-and-fitness/fitness/get-started-safely/blood-glucose-control-and-exercise.html*

[12] Kris Berg, *Exercise Often Raises Blood Glucose in Type 1 Diabetes [Accessed 2015 29 April]; Available from: http://www.diabeteshealth.com/news/exercise-often-raises-blood-glucose-in-type-1-diabetes/*

[13] Fd Turner, D., Luzio, S., Gray, B. J., Dunseath, G., Rees, E. D., Kilduff, L. P., Campbell, M. D., West, D. J., Bain, S. C., Bracken, R. M., Impact of single and multiple sets of resistance exercise in type 1 diabetes, *Scandinavian Journal of Medicine & Science in Sports*, 2015, 1, 25, E99-E109

[14] Bacchi, E., Negri, C., Zanolin, M. E., Milanese, C., Faccioli, N., Trombetta, M., Zoppini, G., Cevese, A., Bonadonna, R. C., Schena, F., Bonora, E., Lanza, M., Moghetti, P., Metabolic Effects of Aerobic Training and Resistance Training in Type 2 Diabetic Subjects A randomized controlled trial (the RAED2 study), *Diabetes Care*, April 2012, 35, 676-682

[15] Diapedia, *Brain glucose metabolism [Accessed 2015 29 May]; Available from:* http://www.diapedia.org/metabolism/brain-glucose-metabolism

Address for correspondence

Aleksandr Agafonov, Department of Computer Science, University of Tromsø - The Arctic University of Norway, Norway
email: agych.alexander@gmail.com, tlf: +4794103421