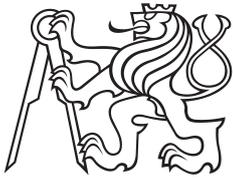


Master's Thesis



Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering

Department of Cybernetics

Record, Representation and Use of Electronic Partogram and Cardiotocogram

Petr Stuchlík

Biomedical engineering and informatics

May 2015

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

DIPLOMA THESIS ASSIGNMENT

Student: Bc. Petr Stuchlík

Study programme: Biomedical Engineering and Informatics

Specialisation: Biomedical Informatics

Title of Diploma Thesis: Record, Representation and Use of Electronic Partogram and Cardiotocogram

Guidelines:

1. Study the issue of a partogram and a cardiotocogram in a clinical environment.
2. Design and implement a structure for recording of the partogram with a link to the cardiotocogram.
3. Design and implement a system for recording of the partogram.
4. Propose a case representation from partogram, cardiotocogram and clinical data.
5. Test created system of the partogram recording and the case representation.

Bibliography/Sources:

- [1] Chudáček, V.; Spilka, J.; Lhotská, L.; Janků, P.; Koucký, M. et al.: Assessment of Features for Automatic CTG Analysis Based on Expert Annotation In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. New York: IEEE, 2011, p. 6051-6054. ISBN 978-1-4244-4122-8.
- [2] Huptych, M.; Chudáček, V.; Janků, P.; Lhotská, L.: An Application for Obstetrics and Neonatology: Electronic Labor Book. In: IFMBE Proceedings: World Congress on Medical Physics and Biomedical Engineering. Heidelberg:Springer, 2012, vol. 39, p. 1257-1259.
- [3] Richter, M. M.; Weber R. O.: Case-Based Reasoning. Springer-Verlag Berlin Heidelberg 2013.
- [4] Wilson, D. and Martinez, T.: Improved Heterogeneous Distance Functions. In: Journal of Artificial Intelligence Research, vol. 6, 1997, pp. 1-34.
- [5] Carter, J.H.: Electronic Health Records. American College of Physicians, 2008. ISBN 978-1-930513-97-6.

Diploma Thesis Supervisor: Ing. Michal Huptych, Ph.D.

Valid until: the end of the summer semester of academic year 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 16, 2015

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Petr Stuchlík

Studijní program: Biomedicínské inženýrství a informatika (magisterský)

Obor: Biomedicínská informatika

Název tématu: Záznam, reprezentace a využití elektronického partogramu a kardiokogramu

Pokyny pro vypracování:

1. Nastudujte problematiku záznamu partogramu a kardiokogramu v klinickém prostředí.
2. Navrhněte a realizujte strukturu pro záznam partogramu s propojením na záznam kardiokogramu.
3. Navrhněte a realizujte systém pro záznam partogramu.
4. Navrhněte způsob reprezentace případů ze záznamů partogramu, kardiokogramu a klinických údajů.
5. Otestujte navržený systém pro záznam partogramu a způsob reprezentace případů

Seznam odborné literatury:

- [1] Chudáček, V.; Spilka, J.; Lhotská, L.; Janků, P.; Koucký, M. et al.: Assessment of Features for Automatic CTG Analysis Based on Expert Annotation In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. New York: IEEE, 2011, p. 6051-6054. ISBN 978-1-4244-4122-8.
- [2] Huptych, M.; Chudáček, V.; Janků, P.; Lhotská, L.: An Application for Obstetrics and Neonatology: Electronic Labor Book. In: IFMBE Proceedings: World Congress on Medical Physics and Biomedical Engineering. Heidelberg:Springer, 2012, vol. 39, p. 1257-1259.
- [3] Richter, M. M.; Weber R. O.: Case-Based Reasoning. Springer-Verlag Berlin Heidelberg 2013.
- [4] Wilson, D. and Martinez, T.: Improved Heterogeneous Distance Functions. In: Journal of Artificial Intelligence Research, vol. 6, 1997, pp. 1-34.
- [5] Carter, J.H.: Electronic Health Records. American College of Physicians, 2008. ISBN 978-1-930513-97-6.

Vedoucí diplomové práce: Ing. Michal Huptych, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 16. 1. 2015

Acknowledgement / Declaration

I take this opportunity to express gratitude to my supervisor Michal Huptych for his useful comments, remarks and engagement through the learning process of this master thesis. I also thank to Petr Olšák for providing the students with this \TeX template. I am also grateful to my partner who supported me on this venture.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

I declare that I have completed this thesis independently and that I have listed all the information sources in accordance with the Guidelines about adhering to ethical principles in the preparation of university theses.

Prague, May 10, 2015

.....
Signature of the author

Abstrakt / Abstract

V této práci opouštíme papírové záznamy porodních křivek a soustředíme se na jejich elektronické zpracování. Cílem je vývoj systému pro zpracování elektronických partogramů včetně odpovídajících datových struktur v relační databázi. Zajímavým použitím získaných klinických dat je možnost jejich použití v systémech podpory rozhodování. Součástí práce je také návrh a testování aplikace pro case-based reasoning na základě klinických dat, partogramu a kardiogramu.

Klíčová slova: Elektronický partogram, KTG, klinický počítačový systém, case-based reasoning, počítačové zpracování partogramu, systém podpory rozhodování

Překlad titulu: Záznam, reprezentace a využití elektronického partogramu a kardiogramu

In this work we abandon paper records with obstetrics statistic and focus on their electronic processing. A goal is to develop a system for electronic partogram processing including relevant data structures in relational database. An interesting way of clinical data post-processing is to use them in decision support systems. A significant part of this thesis is also devoted to development of an application for case-based reasoning with clinical data, partogram and cardiogram.

Keywords: Electronic partogram, CTG, clinical computer system, case-based reasoning, partogram computer processing, decision support system

Contents /

1 Introduction	1	3.5 Application in medical diagnosis	23
1.1 Research of existing applications	1	3.5.1 Electronic health record	24
1.2 Goals specification	2	4 System design	26
1.3 Thesis structure	3	4.1 Requirements	26
2 Obstetrics theory background ..	4	4.2 Use Cases	27
2.1 Partogram	4	4.3 Components and deployment	31
2.1.1 History	4	4.4 ELB data model	32
2.1.2 Applications	5	4.4.1 Event model	32
2.1.3 Partogram related factors	6	4.4.2 Modifications	35
2.2 Cardiotocogram	7	4.5 ELB API	36
2.2.1 FHR variability	7	4.5.1 Architecture	36
2.2.2 CTG related methods ...	8	4.5.2 Components	37
2.3 Clinical data factors	9	4.6 Parto website	40
2.3.1 Length of pregnancy	9	4.6.1 User interface design ...	40
2.3.2 First stage of labor	10	4.6.2 Website structure	40
2.3.3 Second stage of labor ...	10	4.7 CBR	42
2.3.4 Sectio caesarea	10	4.7.1 Application structure ...	43
2.3.5 Blood loss	11	4.7.2 Case structure	44
2.3.6 Induced labor	11	4.7.3 Case description	45
2.3.7 Multiple birth	11	4.7.4 Case solution	48
2.3.8 Gravidity and parity ...	12	4.7.5 Similarity functions	48
2.3.9 Maternal age	12	5 Implementation	52
2.3.10 Admittance reasons	12	5.1 Technology	52
2.3.11 Position of mother	13	5.2 ELB API	55
2.3.12 Fetal position	13	5.3 Parto website	56
2.3.13 Anesthesia and analgetics	15	5.4 CBR application	58
2.3.14 Acid-base balance	15	5.5 Testing and results	58
3 Case-based reasoning	17	6 Data analysis	60
3.1 Introduction	17	6.1 Introduction	60
3.2 CBR process model	18	6.2 Methodology	60
3.3 Case structure	19	6.2.1 Encoding of attributes ..	61
3.4 Similarity	21	6.2.2 Reduction of diagnoses	61
3.4.1 Similarity metrics	22	6.2.3 Tools	61

6.3 Experiments	63
6.3.1 Experiment 1	64
6.3.2 Experiment 2a	66
6.3.3 Experiment 2b	68
6.3.4 Experiment 3a	70
6.3.5 Experiment 3b	72
6.3.6 Experiment 4	74
7 Conclusion	76
References	77
A UML diagrams	83
A.1 Data model	83
A.2 Class diagram of DTOs	86
B Images	88
B.1 Sample partogram scan	88
C Coding of attributes	89
D Experiment samples	91
D.1 Experiment 1	91
D.2 Experiment 2a	92
D.3 Experiment 2b	93
D.4 Experiment 3a	94
D.5 Experiment 3b	95
D.6 Experiment 4	96

Chapter 1

Introduction

For recording of events during labor in many maternity hospitals a paper form with birth curves is used. The purpose of filling these “partograms” is to get an overview of the course of labor, maternal and fetal medication dosages, procedures and potential complications.

The aim of this work is to create a modern tool for computerized recording of these data – an electronic partogram. Electronic documentation brings along a number of benefits: instant access to patient records, sharing information with patients and doctors, more effective diagnostics and potential reduction of medical errors. [1]

Another advantage is the interconnection of data and systems as well as the possibility of subsequent data analysis and evaluation in a variety of applications working with big data. One of such applications in real life is a decision support such as case-based reasoning (CBR).

The ambition is to supplement a database of clinical data with the information from electronic partogram and then try to find similar cases in these data. Based on the similarity of these cases, then we can infer the possible outcomes of new cases.

1.1 Research of existing applications

We base our work on an established database of *Electronic Labor Book (ELB)* by Huptych et al. (2). For the purpose of this thesis, ELB database serves as a base data model and framework that we extend as well as a source of clinical data. The data has been anonymized and was obtained through informed consent from the Department of Gynecology and Obstetrics of University Hospital Brno Bohunice, where the ELB is currently deployed. [2] Part of this work is to design and implement an expansion of the database – a data model for storing electronic partograms.

According to this thesis specification, we analyze an existing solution for CTG signal examination – *CTG website*. This web application was designed and deployed by the Biodat research group in the Department of Cybernetics, FEE CTU.

There are following types of electronic partograms: the first are paper partograms in combination with electronic digital pen (like 3). Another type are dedicated electronic handheld devices similar to tablets, specifically designed to support electronic partograms (like 4). Option is also an electronic partogram as an application for a workstation or a mobile device. As far as we learned, this type of electronic partogram seems to exist for iOS devices only (5).

This work focuses on the third type by implementing a web application for wider range of devices with only a web browser required.

One of the goals of this thesis is to design and test a representation of clinical cases and defined their mutual similarity using CBR. For this purpose, several applications and frameworks have already been created. In our work we try to use one of these instead of inventing a new solution. We prefer open-source solutions, therefore the candidates are for example: *IUC BRF* by Indiana University (6), *jCaBaRe* ([7]), *FreeCBR* ([8]) or *jCOLIBRI* by Complutense University of Madrid (9). We choose the *jCOLIBRI* framework, because it best suits our needs and because it is considered a reference platform for CBR applications development.

1.2 Goals specification

Now we introduce goals of this thesis. The first goal is to **study the issue of partogram and cardiogram in a clinical environment**. To achieve this goal we need to analyze a decent amount of topics and terms from the clinical environment, mainly obstetrics and gynecology. This is covered in the Chapter 2 Theory.

Second goal is to **design and implement a structure for recording of the partogram with a link to the cardiogram**. We analyze available paper partograms as well as the ELB clinical database and the CTG website. We propose a data model based on *Events* in Chapter 4.

With the basic understanding of the topic and with the data structures implemented, we can proceed to requirements analysis of a clinical system. We then **design and implement a system for recording of the partogram**, i.e. the third goal of this work. We propose a modular design and service oriented architecture (SOA) in Chapter 4 and describe its implementation in Chapter 5.

Once we have a working system that can process all the necessary data, we can proceed to the next goal – i.e **to propose a case representation from partogram, cardiogram and clinical data**. Therefore we propose a clinical case description in obstetrics domain and corresponding similarity functions for use in case-based reasoning. We cover this topic in Chapter 3.

The final goal of our work is to **test the created system of the partogram recording and the case representation**. We design and implement an application that uses data from our database, partograms and CTG. This application can transform these data structures into the proposed form of *Cases* and allows to assess their mutual similarity. Additionally we export the result and perform their analysis in Chapter 6.

Note, that additional *implementation goals* (system features and requirements) are introduced in Chapter 4.

■ 1.3 Thesis structure

At this point we provide the reader with a brief overview of the thesis structure.

We start with obstetrics theory research in Chapter 2, including partogram and cardiotocogram (CTG) topics. We continue with case-based reasoning (CBR) introduction and mention briefly the electronic health record (EHR) problematics in Chapter 3. We present a design of the whole system for electronic partogram recording and we decompose it to functional and logic components (Chapter 4). Next we propose a description of used technology and testing in Chapter 5. In the end we design a methodology and perform experiments with the clinical cases and their mutual similarities, the influence of CBR attributes on case clusters homogeneity and intra-cluster variability of case solutions is studied – Chapter 6.

Chapter 2

Obstetrics theory background

In this chapter we deal with a list of physiological and anatomical terms commonly used in obstetrics and gynaecology. We first introduce the key concept of *partogram* and *cardiotocogram* recording. Next we try to describe most of the *clinical data* variables or concepts that we will need to understand during our clinical system design.

2.1 Partogram

“Partogram is a graphical record of important processes taking place during labour including cervix dilatation, head descend and fetus position changes observed during vaginal examination and performed every 1 or 2 hours. Based on partogram deviations from the normal course, actions can be made to regulate progression of the labour, especially by administering medication to enhance uterine activity and when necessary, use birth analgesia as well. With optimal indication and in appropriate time these measures can significantly contribute to the good condition of mother and fetus.” [10]

Although there is no standardized form and attributes in various types of partograms may vary, cervical dilatation and fetal heart rate (FHR) are generally the most carefully observed ones. Among other commonly used (time dependent) variables belong mother blood pressure, pulse and temperature, frequency and intensity of contractions and rotation of fetus.

2.1.1 History

The beginning of partogram goes to fifties, when E. A. Friedman following a study on a significant number of women in the USA and described a normal cervical dilatation pattern. [11] In 1971 H. R. Philpott came with simple but effective graphic labour records and applied them in Central Africa. This “first partogram” was inspiring for other researchers. [12]

In these days partogram is a common tool in developed and developing countries and it is recommended by World Health Organization (WHO). The WHO recommendations contain advices on diagnosis, prevention and treatment of delay in

the first stage of labour. They recommend a four-hour action line for monitoring the progress of labour as well. [13]

■ 2.1.2 Applications

According to Cartmill and Thornton presentation of partogram information influences obstetric decision-making [14] and according to Studd, a partogram can be used to classify normal and abnormal childbirths. [15] There are various studies that proved a positive influence of using partograms – Bosse, Massawe and Jahn claim, that partogram (which they are referring to as cervicograph) can reduce maternal and fetal morbidity and mortality. [16]

According to WHO the critical function of partogram is to *“give early warning that labour is likely to be prolonged”*. [17] In labours where partograms were used, WHO registered a reduction of 3 % in prolonged deliveries and 11 % in interventions. [18]

Other studies showed that the critical factor for length of labor is cervimetric progress, which is one of the most carefully observed variable in partograms. *“An early warning of prolonged labour may be obtained by reference to a line representing normal cervimetric progress of primigravid labour for a given admission dilatation superimposed upon the cervicograph.”* [15] Research on 1000 women in Karachi proved that *“by using partogram, frequency of prolonged and augmented labour, postpartum haemorrhage, ruptured uterus, puerperal sepsis and perinatal morbidity and mortality was reduced.”* [19]

Studies about effect of partogram on rates of Caesarean section, which is among other things related with mentioned length of labor, have an ambiguous results. WHO registered some improvement[18], but on the other hand there are researchers that observed only minimal influence to Caesarean section – for example Seaward from University of Toronto. [19]

A partogram can be used during every stage of labor, but the most frequently used partogram is the one used during the first stage of one. It is also the type of partogram that brings most of the relevant data for computer processing, which is addressed later in this thesis.

But according to Sizer, Evans, Bailey and Wiener the second-stage partogram can be usefull too: *“A second stage partogram is a logical extension of the first-stage partogram and furthers its advantages. Many factors influence progress in the second stage: the size and shape of the pelvis, which may be related to the height of the mother; birth weight; uterine action; soft-tissue resistance; maternal effort; and degree of flexion, caput, and molding of the fetal head. However, these factors may all be resolved into the end result of descent and rotation, which allows simplified assessment and the possibility of graphic representation.”* [20]

They admit that only some women can benefit from the second stage partogram, because most of the women in normal labor deliver quickly. *“In women whose labors are complicated, the benefit of a partogram should be greater.”* [20]

■ 2.1.3 Partogram related factors

We shall summarize other useful factors that come into play when dealing with partograms. The described factors are only a relatively small subset of all possible information a partogram can cover and their total enumeration would not fit inside this thesis. These are only the most common ones that were identified during an analysis of materials provided by the birth center.

Cervix score

Cervix score evaluates the condition of the cervix in late pregnancy. *“One evaluates (by 0–2 points each) the presenting part of the fetus, dilation (degree of opening), cervix effacement, its consistency and localization.”* At the end of pregnancy the total score should be evaluated by 10 points – it means that the presenting part is fully developed, the cervix is dilated cervicis and ectocervix is soft. [21]

APGAR score

One of the basic examinations of the newborn child is determining Apgar score. The method resides in evaluation of five symptoms, which can have value from 0 to 2. The score is determined three times – in the first, fifth and tenth minute after birth. Healthy babies have already in the first minute Apgar score between 7 and 10 points. [22] According to Čech, the value seven already falls into the band of mild or medium birth asphyxia. Values under three points indicate heavy birth asphyxia. Heart rate, breathing, muscle tone, reaction to irritation and skin color are evaluated. [23]

Amnioscopy

Examination of amniotic fluid, or amnioscopy, is used for examination of color and quality of amniotic fluid. An amnioscop is inserted through the vagina and using endoscopic illumination it allows to visually asses the appearance of amniotic fluid. The fluid should be clear. Yellow color indicates hemolytic disease of the fetus, green is a sign of fetal hypoxia. Brown color is found in cases of fetal death. Complications during the procedure include rupture of the membranes, infection, bleeding or causing premature contractions. [23]

Urine test

During admittance a screening of the woman’s urine is performed on protein, sugar or ketones. [23]

2.2 Cardiotocogram

Cardiotocography is used to recording of fetal heart rate (cardio) and uterine contractions (toco). Fetal heart rate (FHR) can be monitored either internally or externally. In external monitoring an ultrasound sensor is put on the woman's abdomen, whereas during internal monitoring an electrode is put on the head of the fetus. A record sample is given in Fig. 2.1.

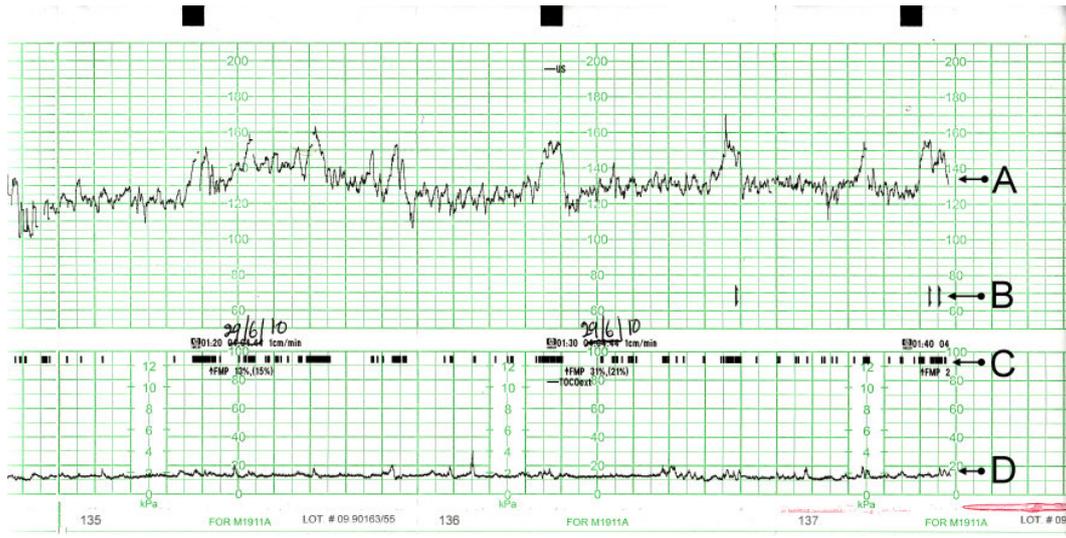


Figure 2.1. CTG record sample; “A: Fetal heartbeat; B: Indicator showing movements felt by mother (caused by pressing a button); C: Fetal movement; D: Uterine contractions”, sample taken from [24].

The uterine activity can be monitored externally and internally. The external tocometry uses a pressure sensor in abdomen area. “*The frequency of contractions and intervals between them can be objectively determined, as well as their duration and fetal movement*” [22], however “*intrauterine tensometry records objective pressure values affecting the fetus.*” [22]

2.2.1 FHR variability

A cardiotocograph records three basic types of frequency variability – long-term, medium-term and short-term frequency fluctuations. Long-term frequency fluctuations include changes in fetal heart rate baseline. Physiological report ranging from 110 to 150 beats per minute is described as normocardia. Over 150 BMP is called tachycardia, whereas until 170 BMP it is classified as subtle tachycardia. It is a suspected report, which can indicate a subtle disorder of oxygen supply. Values over 170 BPM it is a patalogic, severe tachycardia. Values lower than 110 BPM are subtle bradycardia, values under 100 BPM are pathologic bradycardia. [23]

During medium-term frequency variability, transient heart rate accelerations or its decelerations occur. We diagnose an acceleration when there is a heart rate increase of at least 15 BPM with a duration of 15 seconds to 3 minutes. The decelerations is then indicated by heart rate decrease of at least 15 BPM with a duration of 15 seconds to 3 minutes. We distinguish early, late and variable decelerations. Early decelerations occur with the onset and peak contractions, the late ones on the other hand reach beyond the end of the contraction. During excess early decelerations there is a risk of fetal hypoxia, in late deceleration an amplitude needs to be monitored. With the increase of the amplitude, the degree of hypoxia is rising. [23]

Short-term frequency effects are called oscillations or variability and they “*reflect the ever-changing impulses in the sinus node, which origin according to immediate need in the vegetative muscle and they are a reaction of the fetal heart to changes of blood distribution in its peripheral circulation.*” Macrofluctuations or long-term variability tend to have longer oscillation period up to 15 seconds. [23]

There are four variability or oscillation bands. A report is physiological when the band is between 10 to 25 BPM. When the bandwidth is from 5 to 10 BPM, fetal circulation is suppressed. After 40 minutes there is a risk of fetal hypoxia. Band under 5 BPM indicates severe hypoxia of the fetus and birth must be quickly terminated. Band over 25 BPM indicates partial umbilical cord compression. [22]

Cardiotocography is performed already during pregnancy in order to evaluate the uterine motility and fetal hemodynamics. During the delivery at least a 10 minutes long entrance test should be performed. Both stages of labor are continually monitored in high-risk births, intermittent monitoring is possible in purely physiological births. [23]

■ 2.2.2 CTG related methods

When dealing with CTG problematics, two related concepts should be mentioned: ST analysis and FIGO features analysis.

ST analysis

The method of ST analysis of fetal EKG comes from Sweden and is able to detect fetal stress and decrease of oxygen in myocardium. ECG of the fetus is acquired with a spiral electrode on the presenting part of the fetus. The method can detect all three phases of intrauterine fetal distress: hypoxemia, hypoxia and asphyxia. Rise of the T wave is a reaction to the release of adrenaline and indicates hypoxia. Short-term increase in the ratio of the T wave and QRS signals short-term hypoxia. If an increase of the T/QRS line lasts longer than ten minutes, it means that the

fetus is exposed to anaerobic metabolism. If there is a biphasic ST interval, a risk of fetal heart failure rises. It is a pathological condition. [23]

FIGO features analysis

The International Federation of Gynecology and Obstetrics (FIGO) guidelines were introduced in 1987. These general guidelines can be used to improve the results of CTG by additional evaluation of morphological FHR features. [25]

“Attempts to use computer evaluation of the CTG are as old as the guidelines themselves. FIGO features became fundamental in most of the clinically oriented systems and automatically extracted morphological features have been integrated also into automatic systems for CTG analysis mostly for antepartum CTG evaluation.” [25]

2.3 Clinical data factors

In order to design and implement a system in the domain of obstetrics, one should first understand the concepts that the clinical data represent. That is why we first present the reader a list of the most significant variables available in the *Electronic Labor Book (ELB) database* [2] along with an explanation of their significance and impact on result of the labor. These terms will be referred to later in this thesis. Their basic understanding was necessary and allowed us to design the data storage correctly and design a valid case representation.

2.3.1 Length of pregnancy

“The human pregnancy takes 280 days on average since the first day of the last menstruation with variation from 266 to 294. As full term birth it is considered a delivery of the child in 38. to 42. week. Preterm birth is the birth of a baby at less than 37 weeks, while postmature birth is considered the one after completing 42. week of pregnancy.” [22]

The length of pregnancy can have significant effect on both progress and result of the childbirth. The preterm births tend to have higher perinatal and neonatal mortality and morbidity. Common reasons for preterm births are physiological complications on the side of the mothers like adverse obstetrical anamnesis, larger uterine cavity span or uterine anomalies, which also influence the very course of the delivery. [22]

In pregnancies carried beyond the estimated date of delivery the birth can be influenced by bigger size of the fetus, which can lead to prolonged labors, birth injuries

of the mother, head injuries or shoulder dystocia of the fetus. The child can also be in distress because of lack of oxygen or due to compression of the umbilical cord. Postmaturity is also a common indication for an induced delivery. [22]

■ 2.3.2 First stage of labor

The first stage of labor, also referred to as the stage of opening, begins with the first regular uterine contractions. The last phase of this stage is deceleration, during which a gradual dilatation of cervix occurs until the uterine ostium declines. Optimal rate of dilatation is between 3.5 to 5.7 cm per hour. During this stage of labor membranes rupture and amniotic fluid leaks. [22]

This stage is the longest part of the delivery. It takes about 12–14 hours in primiparas (first full term pregnancies) and 6–8 hours in multiparas – the women already gave birth. [26]

■ 2.3.3 Second stage of labor

The second stage of labor begins when the cervix is fully dilated. During this stage the head gradually descends through the birth canal. [22] Before the head reaches pelvic outlet, shoulders are entering into pelvic inlet. The more frontally positioned shoulder enters first. *“The torso of the fetus then delivered without any special mechanism. With the fetus delivered the second stage of labor ends. It takes about 15–20 minutes on average in primiparas and about 5–10 minutes in multiparas”.* [23]

■ 2.3.4 Sectio caesarea

“Caesarean section is the most common operation, which terminates third trimester pregnancy and childbirth. Unlike the vaginal operations that terminate the birth in the end of the second stage of labor, caesarean section is usually performed sooner.” Caesarean section is divided into primary (or planned), which is performed being accessed, for example, because of the illness of the mother or due to pelvic indications, and secondary (or unplanned), when an acute situation that threatens the mother or child occurred during the birth. [23]

The indications of caesarean section include fetopelvic and cephalopelvic disproportion, placenta previa, premature separation of placenta, irregular position of the fetus, breech position, multiple pregnancy and more (see [23]). The common complications of cesarean delivery include bleeding, embolism, organ damage, infection, postoperative hematoma and hernia in a scar. [23]

■ 2.3.5 Blood loss

During physiological labor, the most bleeding occurs at the third stage of labor when the placenta is delivered. Part of the blood loss is also caused by the birth canal injuries. The normal blood loss at natural birth is considered to be about 500 ml of blood. [27]

Generally, a healthy woman can overcome the loss of up to a litre of blood, but (according to the WHO definition) the blood loss over 500 ml during childbirth is a peripartal bleeding, and therefore an episodic and acute bleeding which can lead to serious conditions of the woman. This condition in recent years belongs among the most common causes of maternal mortality in Czech Republic. [28]

“If the blood loss value is about a litre and a half or more, there is a great risk of serious blood coagulation disorders (so called disseminated intravascular coagulopathy), which may have very harmful long-term negative effects on the health of the mother.” [27]

■ 2.3.6 Induced labor

Induction of labor is considered for example when diagnosed with *“posmaturity, diabetes, fetal hypotrophy, hypertension, renal diseases, after amniotic fluid leak over 24 hours etc.”* [22] In some cases the labor is induced in women who underwent caesarean section, uterine myomectomy or have unripe cervix. [22]

Induced labor may entail greater pains, risk of fetal hypoxia and greater risk of slowly progressing labor due to cervicocorporal dystocia. These often lead to the need for surgery. Nevertheless Zwinger claims that *“when comparing larger data, the results of program births and spontaneous ones do not differ.”* [22]

■ 2.3.7 Multiple birth

For the purpose of this thesis and in order to allow better comparability of the results, we decided to take only singleton births into further consideration. With multiple births a larger number of parameters comes into play. *“Generally, multiple births tend to be riskier than the singleton ones. There are more frequent occurrences of premature birth, increase of blood pressure (gestational hypertension), preeclampsia and other complications.”* [29]

For example, the method of childbirth depends more on fetal position in multiple pregnancy than in singleton pregnancy. *“In general we can say, that it is possible to conduct mostly a vaginal method of delivery for twins with the first fetus in head-first position, while in other head positions than headfirst, for triplets and higher order multiple pregnancies it is recommended to perform caesarean section.”* [22]

This is also evident from the latest available data in Czech Republic – while in 2005 there were 16.4 % of caesarean sections of the singleton pregnancies, in multiple pregnancies 69 % of children were born this way. [29]

■ 2.3.8 Gravidity and parity

“Gravidity is defined as the number of times that a woman has been pregnant and parity is defined as the number of times that she has given birth to a fetus with a gestational age of 24 weeks or more, regardless of whether the child was born alive or was stillborn.” [30] Their difference indicates the number of abortions that a woman had.

Knowledge of outcomes of previous pregnancies can indicate progress of current pregnancy and labor and identify risks. Nulliparas have higher risk of difficult and prolonged labour or pre-eclampsia. Multiparas is much more likely to suffer by uterine atony or rupture, obstetric haemorrhage or placenta praevia. [30]

■ 2.3.9 Maternal age

Maternal age, according to many researchers and statisticians, has effect on the course of labor and its complications. According to Main, complications like premature birth, placenta praevia or abnormal fetal position occur more often in women over 35 years. Consequently, a caesarean section is often performed. In older women the second stage of labor is also prolonged and that relates to more frequent need of oxytocine administration. In mothers over 25 years of age an adverse effect on the activity of the myometrium was observed. [31]

Nevertheless, in developed countries, including Czech Republic, the average age of the mothers and nulliparous women rises gradually. *“Until 1977, the highest specific fertility rates were for women aged 20–24. Between 1998 and 2010, the women aged 25 to 29 years contributed the most to the total fertility. During the last two years, the highest fertility rate counts for women aged 30–34 years.”* In this context, the statisticians are also observing a growing number of caesarean sections. [32]

■ 2.3.10 Admittance reasons

In most cases, pregnant women are admitted after they come to the hospital on their own with a suspected birth. *“Most women come on their own according to instructions when regular contractions occur every 10 minutes or when amniotic fluid leaks.”* [22] Another reason for admittance may also be labor pain.

When a woman arrives at hospital admission with genital bleeding, by Zwinger it is an emergency situation which should be dealt with immediately in the delivery

room. [22] A woman may also be admitted because of pre-planned Caesarean section or induction of labor. Another reason for admission is Partus extra muros (i.e. admittance after giving birth outside of the hospital).

■ 2.3.11 Position of mother

The mother may choose to give birth in a number of positions: birth in supine position – designated for the purposes of this study as “standard”, birth while sitting, laying on the side, on a stool, on a bag, standing, kneeling, squatting or in water. According to Zwinger [22] the most common birth positions of the mother are supine or laying on the side. He denotes those positions as beneficial, because the doctor has a good access to the perineum and due to hygiene reasons.

However, there are studies by which births in supine position are less favorable for the mother and bay than vertical positions, such as squatting or standing. *“A statistically significant decrease for the use of medical analgesia and oxytocin was observed in women using the upright birth position. A significantly lower rate of episiotomy was detected in women who delivered in an upright position compared with women delivering supine.”* [33]

■ 2.3.12 Fetal position

We distinguish several fetal positions and the safest one is head-down position, when the baby descends with top of its head down and it performs progression, flexion, internal rotation, deflexion and external rotation correctly. [22] Other fetal positions come with complications of various types and severity, some include a caesarean.

Breech

During frank breech position, which occurs in about 4 % of labors, the fetus does not descend head-first. During complete breech the knees are also flexed. With these positions the delivery can be vaginal, but it often results in caesarean section. There is also a risk of the head getting stuck, interrupting blood circulation between placenta and fetus or placenta compression. These positions pose much greater threat of hypoxia than the head-first position too. *“If we compare the results of deliveries per vaginam for childbirths in breech position with those in head-first position, an increased number of newborns with hypoxia, lower pH, increased hypoxic damage and frequent traumatism is detected.”* [22]

Deflexion

During delivery a situation may occur, where the head will draw itself apart from the chest and *“performs a deflexion along its transverse axis. According to the de-*

gree of deflexion we distinguish the parietal, frontal and facial presentations.” [22]
The most common and mildest degree of deflexion is the parietal presentation, during which the head descends with its fronto-occipital circumference. Due to increased head circumference there is a risk of prolonged labor and difficult internal rotation. If the second stage of labor takes too long or there is a risk of hypoxia, the situation is solved with vacuumextraction (VEX) or pliers. [22]

Stronger and more complicated is deflexion in frontal presentation, when the head descends with its mento-occipital circumference. *“The course of childbirth is difficult and protracted. Without injuring the fetus, the vaginal birth can be performed only for small fetuses or for women with large pelvis. In the best interest of mother and the fetus a caesarean section is indicated.” [22]*

Maximal deflexion occurs during facial presentation, where we can observe a prolonged first and second-stage labor and generally more difficult parturition. Childbirth can be performed vaginally, but it is often necessary to use pliers, more often the caesarean is selected. [22]

Abnormal rotation

Fetal position at which *“the head on the pelvic floor turns itself during internal rotation into the sacrum deepening instead of below symphysis like during the normal birth mechanism.” There is a risk of prolonging the second stage labor and frequent soft tissue tearing. In cases of suspected hypoxia are used pliers or VEX. [22]*

Transverse occiput presentation

It occurs when the head descends without internal rotation on the pelvis floor. In some cases positioning the mother to vertical position or strengthening uterine contractions. In prolongations of the second stage of labor over an hour or during hypoxia, it is necessary to use pliers or VEX. [22]

Straight occiput presentation

The head descends with its sagittal suture into straight diameter of the pelvic inlet. Head and back of the fetus can be turned either ventrally or dorsally. In this position the labor is ended with a caesarean section, because besides prolonged labor there is also a risk head conformation or uterus rupture. [22]

Asynclitism

“Anterior asynclitism arises when the sagittal suture approaches to the promontory and the head of the baby presents by anterior side of the parietal bone. At birth

the head conforms, posterior parietal bone is being subducted beneath the anterior, subdural hemorrhage and central nervous system injuries occur.

Posterior asynclitism is less common than the anterior. The sagittal suture is closer to the symphysis and the head presents by the posterior parietal bone. During distinctive form the vaginal birth is possible only at the cost of head conformation with a high risk of trauma to the fetus. Therefore it indicates birth termination by a caesarean section.” [22]

Cephalopelvic disproportion

Cephalopelvic disproportion (CPD) occurs when the head of the child is much larger than the pelvis of the mother. It occurs in two percent of births, mostly in women smaller than 150 cm. Suspicion is also for primiparas, asynclitism and obstructed labors. The birth ends by caesarean section. [34]

■ 2.3.13 Anesthesia and analgetics

When using analgesia during labor it is necessary that the method does not affect the health of the mother or child, and also the uterine activity. Pharmacological methods include general or local analgesia. From the local methods we can mention needling of the vagina, perineum or paracervical analgesia in the cervix. Major procedure is then epidural or spinal analgesia, when the local anesthetic is injected into the epidural, or into the cerebrospinal fluid. According to Čech [23], administration of epidural analgesia does not extend nor shorten the delivery and it also does not increase the number of obstetric surgeries.

Anesthesia can be indicated either general or regional. With caesarean sections, mostly regional anesthesia (epidural or spinal) is used, because it is more safe for the mother and it also allows direct contact of mother and child. If this type of anesthesia can not be used, the general anesthesia is performed. However there is a risk of aspiration and endotracheal intubation difficulties. [23]

■ 2.3.14 Acid-base balance

The acid-base balance can be examined using the fetal blood either directly during the childbirth or postnatally. The measurement during birth is performed due to abnormal CTG, capillary blood from the scalp of the fetus is sampled. A necessary condition is the cervix dilation of at least 2 cm and amniotic fluid leakage. [35]

Postnatal examination is performed using umbilical cord – the blood is sampled both from the artery and vein. [23] The value of pH is measured, CO₂ level and so-called base excess, a parameter, which is defined as “*the amount of strong acid,*

that we would need to add to the examined sample, in order for its pH to reach 7.4. In metabolic acidosis we would need to add a strong base; the corresponding parameter is referred to as lack of bases, base deficit, BD or (more often) it is expressed as negative BE.” [35]

“A big difference between arterial and venous pH (0.20–0.50) and base deficit (3–11 mmol/l) is probably due to acute asphyxia during birth (e.g. during the umbilical cord complication). In chronic hypoxia pH values are low and the difference between the two vessels is small. Normal values of pH, base deficit, blood gas as well as the definition of acidosis, vary among different centers. The usual limit is pH 7.20 in vein and 7.10 in the artery. Increased mortality and morbidity in neonates and long-term consequences, however, are found only in the arterial values below 7.05.” [23]

Chapter 3

Case-based reasoning

3.1 Introduction

Case-based reasoning (CBR) is generally a method of **computer reasoning** which can be used to find solutions for new problems, given a memory of previous experience and a similarity measure.

Remembering and then reasoning from precedents in order to create a similar solution to a new problem seems like a very natural way of dealing with problems. People deal with many everyday situations using similar principles.

“Consider, for example, a doctor faced with a patient who has an unusual combination of symptoms. If the doctor has seen a patient with similar symptoms previously, he or she is likely to remember the old case and propose the old diagnosis as a solution to this new problem. If coming to that diagnosis was time-consuming in the earlier case, this method is a big savings of time. Of course, the doctor can’t assume the old answer is correct. He or she must still validate it for the new case.” [36]

Richter and Weber [37] provide a sound definition of case-based reasoning using the three words it consists of: “A **case** is basically an experience of a solved problem. This can be represented in many different ways. A case base is a collection of such cases. The term **based** means that the reasoning is based on cases, that is, cases are the first source for reasoning. The term most characteristic of the approach is **reasoning**. It means that the approach is intended to draw conclusions using cases, given a problem to be solved”.

*“A case consists at least of a **problem description** (e.g., symptoms) and a **solution** (e.g., a diagnosis or a therapy). Cases are stored in a database of cases called case base. To solve an actual problem a notion of **similarity** between problems is used to retrieve similar cases from the case base. The solutions of these found similar cases are used as starting points for solving the actual problem at hand.” [38]*

For example, let's apply CBR to a domain of car faults diagnostics. A case (a car problem) in this domain can be *described* by a set (or vector) of arbitrary features like: the manufacturer, car age, fuel level, engine state. The solution consists of

a repair procedure. A concrete case might represent a problem with a 2003 Ford car with full tank and no running engine, with a solution “check the battery”.

Note, that each feature (variable) can be binary, categorical, interval etc. and there is no demand for data types to be the same. There is also no demand for the data to be complete in general and one can implement techniques to handle missing feature values without impairing performance of the system, such as: [39–40].

By constructing a set of problem-solution pairs we obtain a **case base**, a memory that we can use in future to decide new problems and learn from this previous experience.

3.2 CBR process model

The CBR can be viewed as a unified process of four specific “R” phases: retrieve, reuse, revise and retain. A view of the process is introduced on figure 3.1.

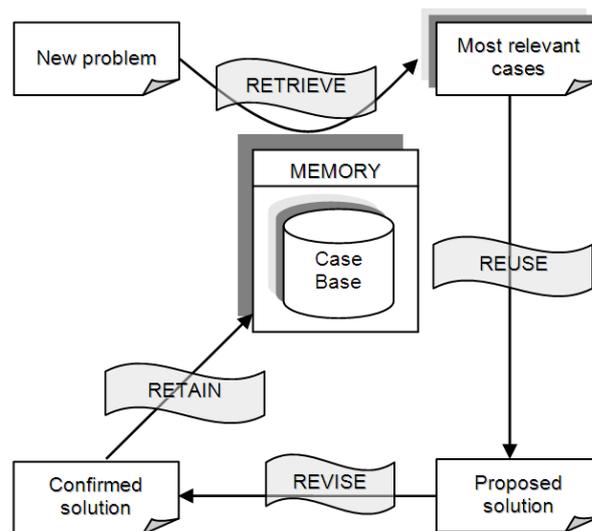


Figure 3.1. CBR process tasks, image taken from [41].

It is worth noting that according to Richter and Weber [37] the CBR can also be viewed as a knowledge-based system. From this point of view, the CBR plays a role of an intelligent system with an independent knowledge base module consisting of four major knowledge containers, each characterized by knowledge it stores (explicitly or hidden): vocabulary, similarity, case base and adaptation container.

Retrieve

By constructing a set of problem-solution pairs we obtain a case base. If a new problem is observed, we can construct a new case (without a solution), often called

a query. Then we use this query to search through the case base and **retrieve** the most similar case and recycle its solution in order to solve our new problem. One might not limit the retrieval result thus obtaining multiple solutions, often sorted by their best match. Multiple techniques of searching the state space (ie. the case base) are used, since: *“retrieval methods are not general; they have to be designed for each system. This is because of the complexity of cases and the inexact matching that CBR implements.”* [37]

Reuse and revise

By recycling the solution we usually mean two specific actions: **reuse**, which is a mapping of the previous solution to the new problem (since the new problem is similar as the previous one). And **revise** - checking if the new solution is applicable. This decision often requires using a substitution, simulation or parameter adjustment (since the new problem might not be the same). [42]

Retain

Once we have obtained a valid solution for the new problem, we can construct a complete case and decide to store it in the case base, making the base more robust using newly obtained experience. This CBR process phase is called **retain** and is often connected with the issue of knowledge maintenance. [37, 43]

3.3 Case structure

The simplest case can be organized using a **flat vector** of attribute-value pairs - like in the previous example with cars. If we use this organization, then the case base becomes a 2D matrix where columns represent features (including a solution) and rows are the individual cases (or vice-versa). An example of such matrix is presented in Table 3.1.

manufacturer	age	fuel	engine running	lights working	diagnosis
Ford	3	10	false	false	check battery
Skoda	4	5	true	false	replace fuse
Volkswagen	1	0	true	true	take fuel
Ford	3	25	false	true	check oil

Table 3.1. An example of flat structure case base.

We can use flat case structure if there are no relationships between the cases (such that we know of or want to represent). Such organization usually simplifies the

design and implementation and is suitable for a small number of cases. [37] By design, flat structures use simple data types of attributes like strings, numbers or boolean values only. Therefore their use in the real world is somewhat limited.

In the real world applications, the cases are more often organized into structures as **hierarchies as networks**. These representations still use attribute-value pairs, however there are various relations linking the features together. Given the nature of these relations, the case structures can be: object-oriented, trees, graphs, hierarchies and taxonomies or set-oriented. [37]

An example of object structure is indicated in Fig. 3.2. Object structures are easier to understand, more compact and allow easier representation of certain aspects. [37]

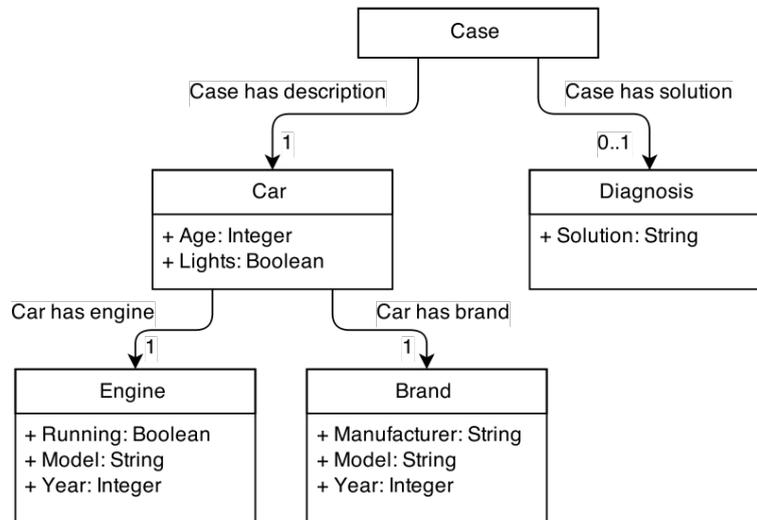


Figure 3.2. Case object structure example.

Graph and tree cases are based on constructing and annotating graphs (both directed and undirected), where vertices represent concepts and edges their relations. For example an ontology of concepts connected by semantic relations can be used directly in CBR with graph case structure. In many cases graphs are underlying structures of other representations and they can benefit from efficient computation.

Choosing a concrete structure depends on requirements of the domain, because we generally want to use the same structure as the data we have available.

When designing a case structure, it is advisable to include additional concepts that might represent some hidden knowledge. An example of such knowledge is a difference of two different attributes “income” and “spending”. Both features are known to the system, but it is the knowledge about their comparability using

difference, that might improve efficiency of the system. These attributes are called *virtual concepts*. [37]

3.4 Similarity

“The purpose of retrieval is to retrieve the case from the case base (i.e., a candidate case) that is so similar to a given new problem that their solutions can be swapped. One of the implications of this concept of similarity within the CBR methodology is that CBR’s similarity is not a general concept, but a polymorphic concept that varies for each case base.” [37]

In order to obtain good match during the retrieval phase of CBR, a **similarity function** need to be defined for each concept we want to compare. For structured cases this means defining similarities for both simple and complex attribute types. The similarity function should be a metric (or generally a distance function) which takes two arguments and returns a similarity measure, usually a value between 0 (completely dissimilar) and 1 (the same).

Furthermore it is useful to be able to scale the relative relevance of each attribute and give greater importance to attributes which play a larger role. This can be done by specifying **weight** for each attribute.

Richter and Weber [37] provide a definition 3.1 of global similarity using local similarity functions and attribute weights:

Definition 3.1. Given arbitrary domains: $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_n)$,

attributes: $Attribute_1, \dots, Attribute_n$,

local similarities: $sim_1(x_1, y_1), \dots, sim_n(x_n, y_n)$; $x_i, y_i \in dom(Attribute_i)$

and weights: $\omega_1, \dots, \omega_n$.

Overall (global) similarity:

$$\sum_{i=1}^n \omega_i sim(x_i, y_i).$$

CBR does not say anything about the origin of the weights that reflect the importance of each attribute. Weights can be derived from one’s knowledge of the domain or using some techniques of statistics and unsupervised machine learning like sensitivity analysis [44], neural networks [45], clustering algorithms [46] or linear programming [47].

■ 3.4.1 Similarity metrics

Many elementary measures of metric similarities (or distances respectively) have been discussed in mathematics. These can be applied directly if we are using attributes with numerical values. In case of symbolic values (i.e. categorical variables) we must first implement numerical coding. [37]

“There is a huge number of simple similarity measures that are used in attribute-based CBR. Often, they differ only in the choice of certain coefficients. This choice is motivated by the needs of some specific application. Mostly, the context gives the required insight into use. However, the choice is mostly ad hoc and there is no principle behind it.” [37]

The most common metrics include: Manhattan distance, Euclidean distance, squared Euclidean or Minkowski distance. [48] We give only a brief review of these functions.

Manhattan distance, also called *city block* distance, represents a distance between two points in a grid, where one can only travel using horizontal and vertical paths. Therefore the distance is a sum of the absolute differences of the points projected onto the coordinate axes:

$$dist(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n |x_i - y_i| \quad (1)$$

Euclidean distance represents the most common straight line distance between two points. It is defined as the root of the sum of squared differences between two objects:

$$dist(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2)$$

The metric can be squared in order to give (non-linearly) more similarity to objects that are closer together in the Euclidean space.

By generalizing the previous and introducing a parameter p we can derive **Minkowski distance**:

$$dist(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}} \quad (3)$$

For $p = 1$ we get the Manhattan distance, for $p = 2$ it becomes the Euclidean distance. A special case of Minkowski distance can be obtained with p reaching ∞ (using a limit), resulting in maximum value distance, also called *Chebyshev distance*. [48]

Cosine similarity

“The cosine similarity is a well known vector similarity measure that measures the angle between the two vectors, and it is commonly used in text mining and computer vision. A disadvantage of the similarity measure is that it does not differentiate between parallel vectors with differing magnitudes.” [49]

This function can be used to evaluate similarity using the “shape” of time series (or sequential data in general). The corresponding formula is given below:

$$sim(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|} = \frac{\sum_{i=1}^n x_i \times y_i}{\sqrt{\sum_{i=1}^n x_i^2} \times \sqrt{\sum_{i=1}^n y_i^2}} \quad (4)$$

Similarity of sets

Similarity measure of cases described using finite set structures can be computed using a statistic called *Jaccard index*. Developed by Paul Jaccard in 1901, this coefficient measures the similarity as the cardinality of the intersection over the cardinality of the union. The corresponding formula is given below: [50]

$$sim(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} \quad (5)$$

3.5 Application in medical diagnosis

CBR application in diagnosis involves solving analytic tasks that are very similar to classification and prediction. *“These reasoning tasks all attempt to associate an instance with a class. Classification does it in its purest form, while diagnosis and prediction are variants. Diagnosis associates an instance with a class that implies a fault, a problem, or a disease. The relationship between these outcome classes (e.g., fault, disease) and the instance are commonly provided by symptoms, and they tend to have an aggregated effect that characterises the outcome.”* [37]

A medical diagnosis task can be also viewed as a classification in general, but instead of searching for a concrete diagnosis, we usually want to find a solution

that treats the patient’s condition. Therefore we omit diagnoses in CBR solutions and present only a concrete remedies, procedures or therapies. [37]

Medical diagnosis applications of CBR often involve use of multiple different CBR models and case representations. Imagine for example a task of identifying pathological cases on x-ray images (i.e. querying a case base using image data) or various sensor signals similarity assessment - both these problems can be solved using CBR, but they require a slightly different approach and techniques when designing the CBR system when compared to more common, feature-value based tasks. In this thesis we focus primarily on the latter - representation of patient’s health records and comparing them according to similarities.

■ 3.5.1 Electronic health record

In this section we present the reader with a brief introduction to Electronic health record (EHR) problematics, because the resulting CBR case representation is greatly attached to this topic. We shall provide some basic concepts that we will refer to in the next chapters, especially during the design phase.

The EHR history dates back to the late 1950s, when the first reports of using computer processing of clinical data appeared. Hospital information technology applications range through a wide spectrum of systems like master patient index, pharmacy and radiology information systems, picture archiving system, laboratory information system and more. A satisfying definition of the EHR concept however was not established before 2003. [51] A report “Key Capabilities of EHR Systems” by the Institute of Medicine [52] states:

“An EHR system includes: 1) longitudinal collection of electronic health information for and about persons, where health information is defined as information pertaining to the health of an individual or a health care provider to an individual; 2) immediate electronic access to person- and populationlevel information by authorized, and only authorized, users; 3) provision of knowledge and decision-support that enhances the quality, safety, and efficiency of patient care; and 4) support for efficient processes for health care delivery.”

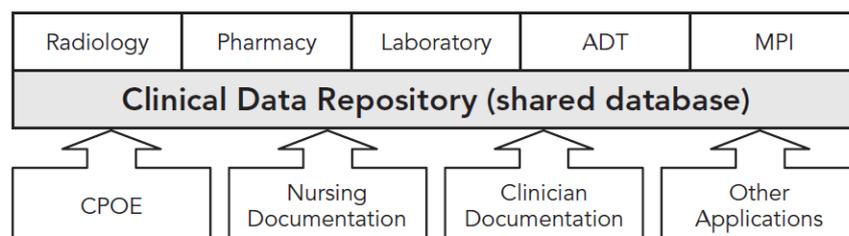


Figure 3.3. EHR schema overview, image taken from [51].

In the recent years there have been significant trend of EHR standardization and interoperability involving organizations like Health Level 7 (HL7) [52]. This standardization is crucial for successful interoperability of clinical systems, because in clinical institutions many IT systems are commonly used (often by different providers). [51] An overview of a standardized EHR with unified database design and integrated components is given in Fig. 3.3.

Chapter 4

System design

In this chapter we deal with design of a system for recording of the partogram with link to the CTG and clinical data. This design extends approach, which was introduced in [2]. We use Unified Modeling Language (UML) as a standard application and system design tool.

We start with a summary of Requirements and their mapping into series of Use Cases. A section presenting the deployment model of the resulting system follows.

Finally the tasks and goals of this work are divided into three main components: API, website and CBR application. For each of these components, a section is dedicated to describing its design further. That includes Entity Relationship (ER) model, Class Diagram and component diagram.

Note, that we did not want to dive into great technical details in this part, so more granular diagrams will be provided in the Attachments only. These include Entity-relationship (ER) model and Class model.

4.1 Requirements

With respect to this thesis goals and after analysis of the problematics described in previous chapters, following requirements for the system were determined:

- R1. The system will allow users to record (create and modify) partograms.
- R2. The partogram should be linked to the existing CTG system, this link shall be realized in data model as well as in the user interface.
- R3. The system will store the data into a derived form of existing database of clinical data (i.e. ELB).
- R4. The system will enable further processing of the data.
- R5. The system will provide a component for case representation and CBR.
- R6. The CBR component will be based on partogram, CTG and clinical data.
- R7. The CBR component will allow best match case retrieval given a query.
- R8. The system will allow only authorized access to all the data available.
- R9. The users can act at different role(s) in the system, an existing system of roles should be used.

R10. According to the latest trends, the system should be compatible with widest possible range of operating systems and devices (including mobile access).

4.2 Use Cases

After requirements analysis, we identified a total of eight use cases and three major actors of the application interacting with the use cases. Use case and actor relations are presented in Fig. 4.1. We shall now introduce each actor and use case briefly.

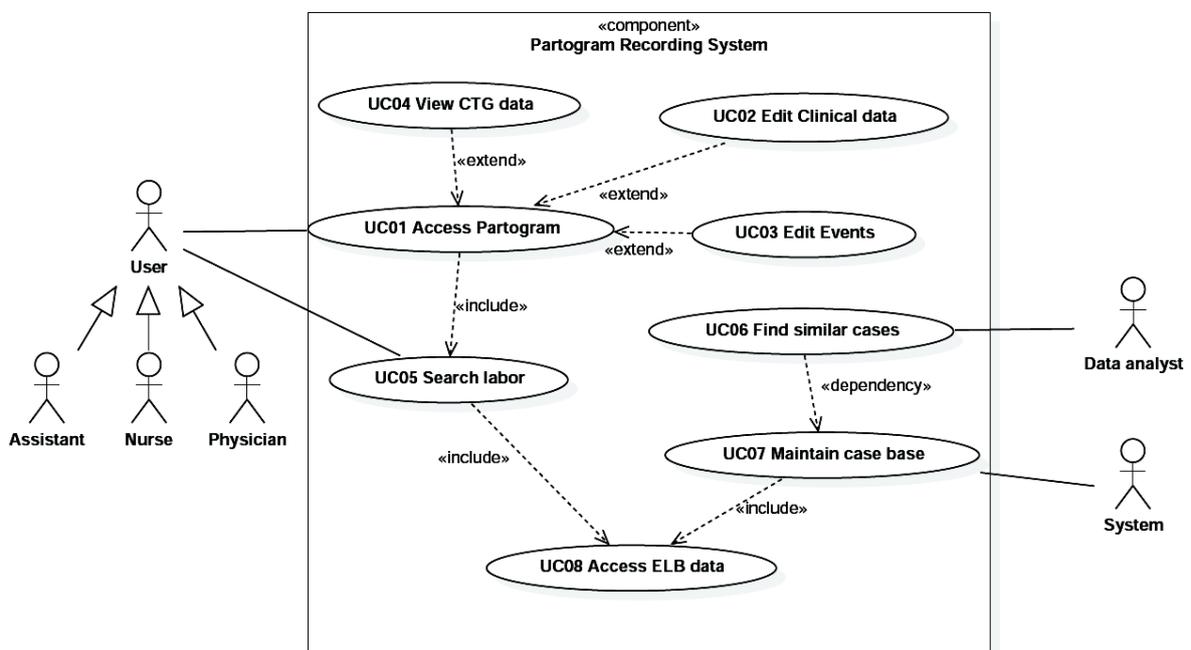


Figure 4.1. Use Case diagram of the system.

User

User is a generic system actor authorized to access the system for recording partogram data. With respect to the existing roles defined in ELB, this user can either be a *Physician*, *Nurse* or *Assistant*. Although these actors share the same use cases, in the system their role should be always recorded in order to preserve the RIM statement principle defined in Section 3.5.1.

Data analyst

Data analyst is an actor whose only purpose is to work with clinical cases and assess them using CBR similarity functions. Therefore it is an actor that is shielded from the partogram-recording use cases.

System

System acts on defined conditions (e.g. repeating time intervals, triggers) and runs complex tasks, that often take longer time to complete and do not require direct user interaction.

UC01 Access Partogram

This basic use case allows the *User* to access the electronic partogram. It is mapping requirements R01, R03, R08 and R10.

Main scenario:

1. The scenario begins with *UC05 Search labor* when a labor is found.
2. The *User* performs an action to open electronic partogram of the given labor.
3. The system loads necessary data from the database and presents the user with a view of electronic partogram.
4. The *User* fills in basic partogram data.
5. The system saves the data into database.
6. The scenario ends.

UC02 Edit Clinical data

Use case extends UC01 Access Partogram in a way that a special form for clinical record of the mother, admittance and newborn is provided. Use case is mapping requirements R03 and R10.

Main scenario:

1. The scenario begins anytime during *UC01, step 4*.
2. The *User* can fill in special forms of the clinical record.
3. The system stores these data in the database.
4. The scenario ends.

UC03 Edit Events

Use case extends UC01 Access Partogram in a way that a dedicated user interface is provided for the *User* to annotate basic partogram data (curves). We will refer to these annotations as “events” from now on. Use case is mapping requirements R01 and R10.

Main scenario:

1. The scenario begins anytime during *UC01, step 4*.
2. The *User* can create annotations of the partogram data.
3. The system then preserves these annotations along with the partogram data in database.
4. The scenario ends.

UC04 View CTG data

Thanks to this use case the *User* can view a link to the CTG data (if present) directly in the Partogram application view. Use case is mapping requirements R02 and R10.

Main scenario:

1. The scenario begins anytime during *UC01, step 3*.
2. The system checks if there are CTG data available for the given record.
3. The system provides an indication, that CTG data are present. The system puts this indication in correct context with the partogram time course.
4. The *User* can use this indication to navigate directly to the CTG application component view.
5. The scenario ends.

UC05 Search labor

The *User* is able to search ELB labor records using filters, sorting and pagination. Use case covers requirements R03 and R10.

Main scenario:

1. The scenario begins with authorized *User* that wants to look up a record.
2. The *User* navigates to the library of ELB records.
3. The *User* fills in search query parameters.
4. The *User* submits the search form.
5. The system processes the input and looks up the matching records using *UC08 Access ELB data*.
6. The system renders the result.
7. The scenario ends.

UC06 Find similar cases

It is possible to search similar cases (k nearest neighbors) from the case base. Use case covers requirement R07.

Main scenario:

1. The scenario begins when the *Analyst* invokes the CBR application.
2. The *Analyst* forms a CBR query description.
3. The system performs K-NN search using the query and the case base.
4. The system returns the search results.
5. The scenario ends.

UC07 Maintain case base

The case base needs to be created and maintained using clinical data, partograms and CTG. In the simplest case, this involves loading the data only once and creating the cases. However in order to keep the base updated, there should also be an automatic mechanism that updates the cases in the base, possibly aggregating case clusters using the cluster centroids.

Use case covers requirements R04–R06.

Main scenario:

1. The scenario starts regularly with a given time period (e.g. once a day).
2. The *System* invokes a CBR procedure that performs case base maintenance.
3. New data are loaded using *UC08 Access ELB data* and cases are created (including both description and solution parts).
4. The case base is created.
5. The scenario ends.

UC08 Access ELB data

This is a task that is common for all the other cases, since each action in the system requires an access to clinical database. Use case therefore covers all the requirements.

Main scenario:

1. The scenario begins when a use case actor needs to work with patient's EHR.

2. The actor forms a search query and submits it to the system (this is achieved using a common API).
3. The system performs the search and returns the result.
4. The scenario ends.

4.3 Components and deployment

Both the model of system components and the deployment model are usually placed in final parts of the technical documentation, but we provide this big picture here as it should help the reader to understand the whole concept sooner. For this reason an unorthodox “UML inversion” is used in this chapter.

We proceed to decomposition of the original problem. We have already mentioned, that there were two systems already running which we could use, extend and build upon: **ELB application** and **CTG website**. The ELB application handles EHR for patients and therefore contains a solid database of clinical data. The CTG website provides CTG user interface (UI) and it is loosely coupled with ELB already.

According to the requirements and use case model, a total of three components were identified: a data retrieval and persistence system providing CRUD functionality, an application client component providing rich user interface for partogram recording and a final CBR component providing case representation and similarity based data retrieval. We will refer to these components as **ELB API**, **Parto website** and **CBR application** respectively.

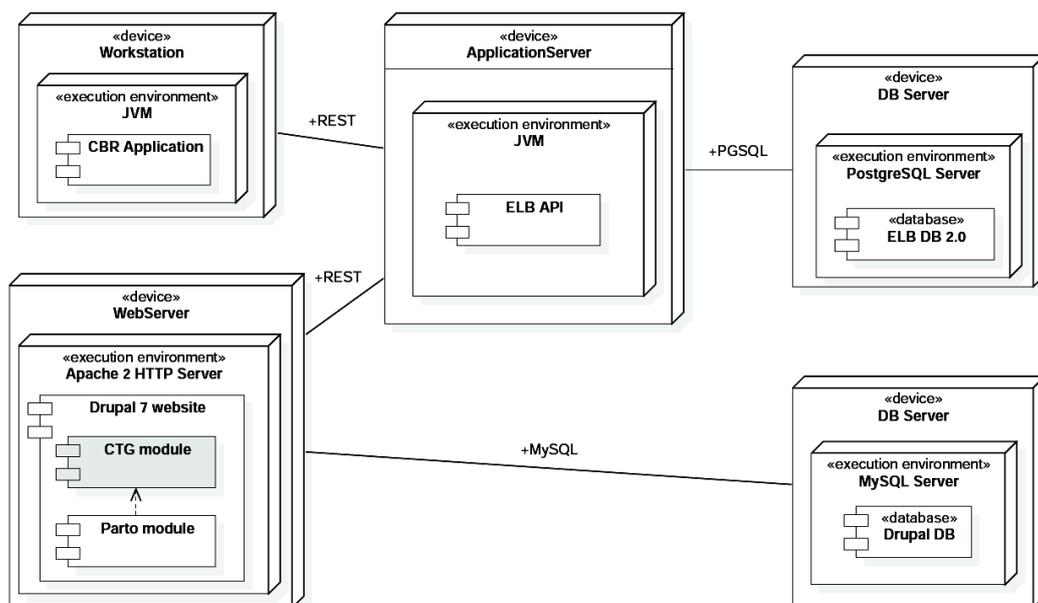


Figure 4.2. Deployment diagram of the system components.

We have then decided to design following architecture and implementation workflow:

- 1) First we **update the data model of the original ELB** in order to meet requirements for the data storage.
- 2) With the new database up and running, we **design and implement the ELB API as a REST service**. Only the service shall work with the data storage directly and should serve as a data resource mediator for the rest of the components.
- 3) We then **extend the CTG website using new Parto module**, thus implementing Partogram recording features. The module shall implement the REST API provided by **ELB API** web service in order to access and store the clinical data.
- 4) Finally we **design and implement the CBR application**. The application shall also implement the REST API, but will require only a read access to the data (since we are only concerned with data retrieval in CBR).

This architecture is also shown in Fig. 4.2.

4.4 ELB data model

With the task of electronic partogram recording we need an underlying data model to work store data in. This is the first thing we design, since all the applications will be designed to work with these data.

We start with analysis of the current ELB relational (PostgreSQL) database (described in 2). The model contains a total number of 41 tables. These tables represent major entities like *Mother*, *Admittance*, *Labor*, *Newborn*, *Personnel* etc., but also enumerations like *Sex*, *Fetal position*, *Diagnosis*, *Intervention* and others. Some of these tables are also purely relational tables (managing many-to-many relations).

It should be noted, that the existing data model is very well designed with respect to the *EHR* concept and *RIM* ideology and while updating this schema, we tried to continue these principles and maintain both structural and principal compatibility.

4.4.1 Event model

After we had analyzed paper partograms, we found out that each procedure recorded there can be represented by a general “Event” in time. Our data model design is based on *Event model* implementation, which has already been introduced in [53]. In this section we design such a data model for use in electronic partogram recording.

Each *Event* is an entity that is defined by following attributes:

- **Time** – it identifies a time instant, during which the *Event* occurred. We further extend this to allow specification of both start and end time of *Events*, that are not instant (with respect to the partogram time resolution, i.e. tens of minutes). There is also a relation specifying time, when a change occurred during the pending *Event*.
- **Type** – *Event type* identifies what kind of *Event* occurred. There are five different types of *Events*, which will be introduced later.
- **Subject** – it is a parent entity for *Newborn* and *Mother*.
- **Author** – identifies *Personnel* who created the *Event*.
- **Labor** – providing a context for the procedure.

Additionally, each *Event* can have a note attached to it.

Event types

The *Event types* are used to distinguish between different kinds treatments or clinical procedures, that are recorded in partograms. They carry information about what happened, whereas the *Events* say when it happened. Using following five *Event types* we can cover all these procedures:

- **Medication** – identifying types of various drugs administered (e.g. oxytocine, nalbuphine, entonox)
- **Measurement** – storing measured values (e.g. blood pressure, pulse, temperature)
- **Observation** – recording observed facts (e.g. cervix score, contractions)
- **Diagnosis** – identifying problems, conditions or diseases (e.g. preeclampsia, abruption of placenta, premature birth).
- **Intervention** – holding information about procedures performed during labor (e.g. caesarean section, forceps, epiziotomy).

We will call instances of these *Event types* the *Event subtypes*. In this terminology, the oxytocine is an *Event subtype* of *Event type* medication, which can identify an *Event* where 1 unit of the drug is administered. With the previous statement we get to the next topic that needs to be addressed – *Attributes* and *Units*.

Event attributes

It should be clear, that each *Event type* is different and covers different kind of procedure. This difference comes into play when we want to record all the information that the *Event* carries. For example, when measuring body temperature, we need

to store the measured value somewhere. But when preeclampsia is diagnosed, we don't need any additional attributes or parameters.

Naturally, this difference is also observed among the *Event subtypes* of the same *Event type*. Again, there is a value to be measured in body temperature, but there are two values to be measured in blood pressure (systolic and diastolic pressure). These attributes have their values inserted arbitrarily (it is the author's responsibility to provide the correct input), while others (like cervix score or apgar score) provide predefined options that the author can choose from. This feature is also implemented in the data model.

Attribute units

Each *Attribute* can have additional relations to entity *Unit*. For example, some medication dosage is in IU (International Unit), while others are in mg or ml. The author of the *Event* chooses the *Unit*, when there are more *Units* possible. There are also *Attributes*, that do not specify any unit at all (e.g. apgar score).

Attribute templates

To enhance user-experience (UX) of the clinical applications, we design an attribute-template system for *Event subtypes*.

The idea is that if a user would fill in some *Event attributes* repeatedly using the same values (including *Units*), he or she can prepare an *Attribute template* ahead. Then the next time the author creates this *Event subtype*, he or she can choose to use values from the template. Such operation is generally much faster than filling the complete form.

Signals

We define the *Signal* as a special (sub)type of *Measurement*. It is used to hold data about signal measurements (like FHR). Due to large data sizes, the signals are usually stored on hard drive. But storing their metadata in the database can be useful, since we can work with each signal without the necessity to read the files on drives.

These metadata contain information about time interval, signal type, sampling frequency and sensitivity, but for each signal, other attributes like baseline or variability can be recorded as well. Signals we work with often come in segments, therefore we define the attributes for each segment separately.

■ 4.4.2 Modifications

We have identified following issues in the model, that need to be resolved in order to support partogram data storage:

- 1) **Subject** table needs to be created with corresponding relations to *Mother* and *Newborn*.
- 2) **Subject type** table is therefore needed to distinguish *Subjects*.
- 3) **Two blood type** tables (AB0 and Rh) need to be created and connected to the *Subject*. This information is filled in partogram header.
- 4) **Event** table is needed to hold information about each procedure.
- 5) **Event type** table is a common ancestor table for the five major types: **Diagnosis, Intervention, Observation, Medication** and **Measurement**. The *Intervention* and *Diagnosis* tables are already present in the current ELB data model and they needed only minor modifications. The other three tables need to be created from scratch.
- 6) **Event change** table holds information about a change that occurred for a particular *Event* at given time.
- 7) **Parameter** table stores enumeration of all available attributes (parameters) and is connected using many-to-many relation with *Event type*.
- 8) **Unit** table stores enumeration of all available units. It uses many-to-many connection with *Parameter*.
- 9) **Parameter option** is used for attributes, that allow predefined values only. It uses many-to-many connection with *Parameter*.
- 10) **Event parameter value** is a table for storing the actual values of *Parameters* and optionally *Units* as well. It is connected with *Parameter*, *Event* and *Unit*.
- 11) **Parameter template** is a table for storing attribute (parameter) templates. It is connected with *Parameter*, *Event type* and *Unit*.
- 12) **Signal** is created to manage information concerning signal data. Signal management is further decomposed in four additional self-explanatory tables: **Signal type, Signal segment, Signal attribute** and **Attribute type**.

The described changes result in a data model update with 60 tables in total.

Other minor adjustments were made to some tables (namely *Labor*, *Newborn* and *Admittance* that facilitate changes in the data model described in the list above. We will not dwell on these details further – the complete data model is given in the Attachment A.

4.5 ELB API

This section is dedicated to Electronic labor book API web service description. We first introduce an overview of the application as a whole as well as some core concepts that the application builds upon. Then we describe the individual components that act as logical and functional units of the API.

4.5.1 Architecture

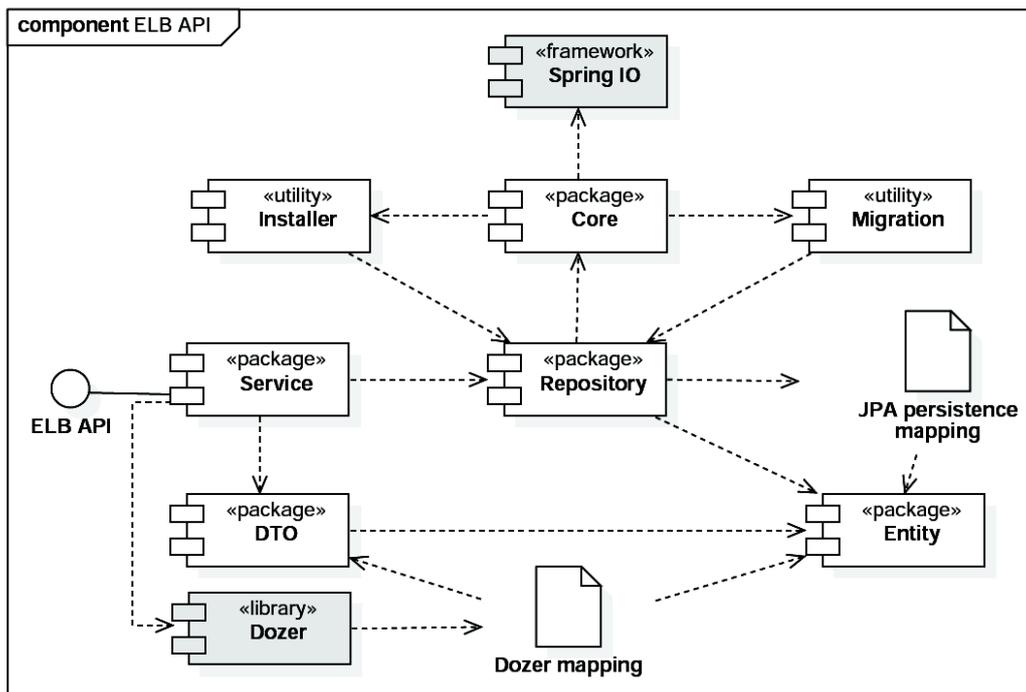


Figure 4.3. Component diagram of ELB API web service.

The ELB API application is designed to work as a web based server with exposed REST interfaces. The interfaces provide the implementor clients with an indirect access to clinical data. Whenever a client connects to the API, its request is processed using a REST controller which invokes proper methods in a CRUD repository. The repository then loads the data from a database (described in the previous section). Once the data are loaded, they are propagated back through the controller and to the client.

This procedure is also designed to work in situations when the client wants to store data (create or update) in a similar way.

The ELB API therefore stands between data consumers and the database – it acts as a mediator. This design pattern is called *Service-oriented Architecture (SOA)*

and is very well established in modern application design. [54] The main benefit here comes with decoupling dependency on complicated database transaction logic and the CRUD interface. The client application can easily invoke methods to create/read/update/delete data without the need of knowing anything about how the data is stored.

This fact allows us to keep the clinical database structure updated without the need to notify all the web service clients about any changes (as long as no changes in the declared interface of the service are made). We can even change the storage from one database provider to another and the clients would still work the same way.

Other benefits come with performance optimizations. Since the API is a central point of data handling, we can involve techniques like data caching or load balancing directly in the web service to fix the most bottlenecks.

Finally, the web service also ensures simple, yet effective security measures. There is simply no other way that any client or third party would access the clinical database than via ELB API. And because each request a client sends to the API must be validated and authorized, we know that each transaction with the clinical data has been validated and authorized as well.

The ELB API architecture is presented in Fig. 4.3. In the following section we shall describe each component in more detail.

■ 4.5.2 Components

Core

This component represents only the basic functionality that is needed to run the application. The idea is to use Spring framework's Boot (described later in chapter 5) in order to start a standalone web server.

The server handles all the resource management as well as life cycle management of the other components, leaving only a few responsibilities to the Core. These include: starting and stopping the web service, invoking the Installer component and starting a migration process when necessary.

Installer

Installer is a utility component used to fill the database with enumerations during the first application run. It will install all the auxiliary data structures like units, event types and signal types.

Migration

This is a utility that allows to perform a batch migration of signal data from a given resource (like a 3rd party API) into the database.

Repository

The Repository package contains a collection of our *Data Access Objects (DAO)*. The main idea behind our repositories is to have exactly one responsible repository class per entity. These classes provide data manipulation interfaces for CRUD operations and pagination and sorting support. They act as a mediator between the domain (entity) and data mapping layers.

We will be using *Java Persistence API (JPA)* which allows us to focus more on the application logic and not to put much effort on writing database queries and old-school wiring. Additionally, by extending *Spring CrudRepository interface*, the complexity of a DAO class is further simplified since each repository is basically a list of interfaces with method headers only – the rest is generic code from Spring framework. An overview of the Spring Repository model is given in Fig. 4.4.

This topic will be covered more thoroughly in chapter 5 Implementation.

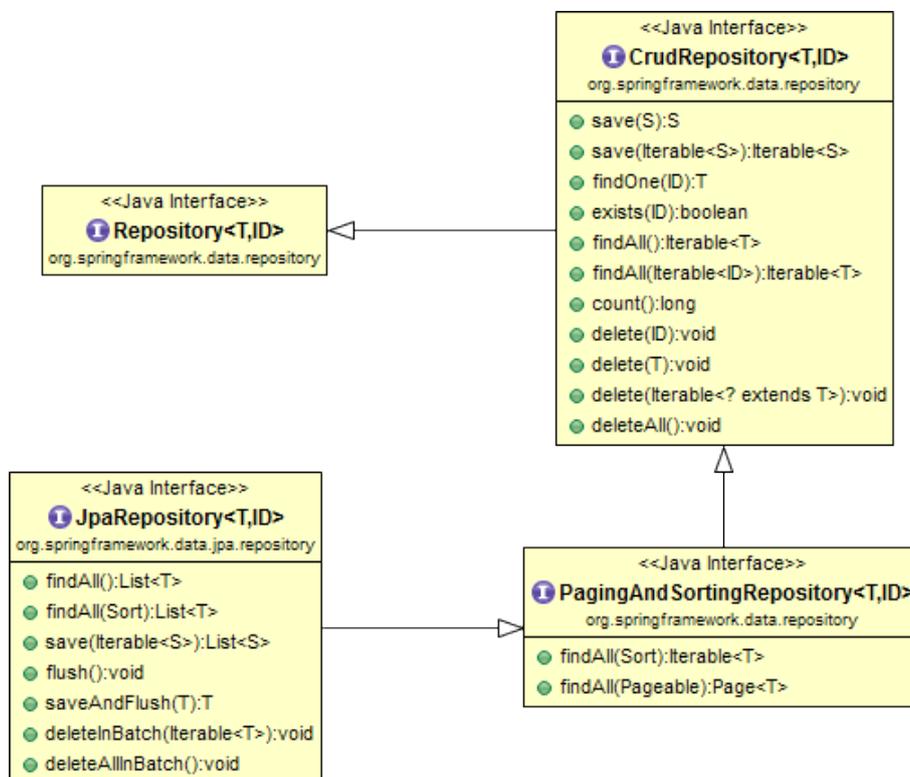


Figure 4.4. Class diagram of Spring framework Repository interfaces, image taken from [55].

Entity

Entities are *Plain Old Java Objects (POJO)*, that represent records in the database tables, described in previous section. As mentioned, we are using JPA, therefore for each database table we create an entity Java class (with exception for some tables which store many-to-many relations only) and this mapping is automatic for classes with the same name as their respective database tables. Each entity then contains a field per database table column.

The entities are annotation driven, which means, that we only need to add `@Entity` annotation before each entity class definition (there is no inheritance). JPA can handle complex data model relations like generalization, which are also used in our model. This is solved simply by entity class inheritance and including an `@Inheritance` annotation to the superclass.

An overview of database tables was already provided as well as their relations, therefore we can omit the model of Entity classes – its structure mirrors the ER model very closely.

DTO

Data transfer object (DTO) is an object that is specially designed to transfer data between two endpoints, usually between web service and its client. We do not need to carry all the entities over the network. In addition we also don't need to carry some of the data fields in entities. For these reasons we define a layer of DTOs which carry only the data we need, thus saving expensive communication calls.

When a DTO pattern is involved, a mapping of business domain (entity) data in some form is necessary. We have decided to use *Dozer*, which is a Java Bean to Java Bean mapper library that recursively copies data between two (different) object types. The complete class diagram of the DTO package is included in the Attachment A.

Service

Components that expose application interfaces as web services reside in the Service layer. It includes Java classes that use Spring `@RestController` annotations, which allows their methods to be mapped to given URLs.

Since we design the services using *Representational state transfer (REST)* architecture, each web service method is mapped to one of the HTTP methods: GET (returning data), PUT (replacing data), POST (creating data) and DELETE (deleting data). [56]

There are four web services in the ELB API:

1. *GreetingService* with the only purpose of the web service health status monitoring.
2. *FolderService* provides read/write access to *Folder* DTO thus being the basic resource point for the clinical data management.
3. *EventService* is an interface providing read/write methods for *Events*, *EventTypes* and *EventParameters*. These methods are needed for partogram data management.
4. *PersonnelService* takes care of Personnel data loading. This service is directly routed to autocomplete function handler in the Parto website (described in the next section).

■ 4.6 Parto website

As mentioned, the website provides accessible user interface for partogram recording. This component is built upon *Drupal 7 Content Management System (CMS)* website, which already provided a CTG component.

In this section we first present the partogram recording UI design. Then we introduce structural website components and their role in partogram recording.

■ 4.6.1 User interface design

We have analyzed the available paper partograms (see Attachment B) and summarized their key features in a form of schematic UI. The UI design is shown in Fig. 4.5.

■ 4.6.2 Website structure

The CTG component modules were analyzed and new partogram modules were designed. The resulting structure is shown in Fig 4.6. Description of each website component follows, starting where we left in the ELB API – the REST client.

REST client

In order to communicate with the ELB API we need a client that implements its interface. This Drupal module is a standalone component without any other dependencies. For communication with the web service it only uses standard Drupal 7 APIs.

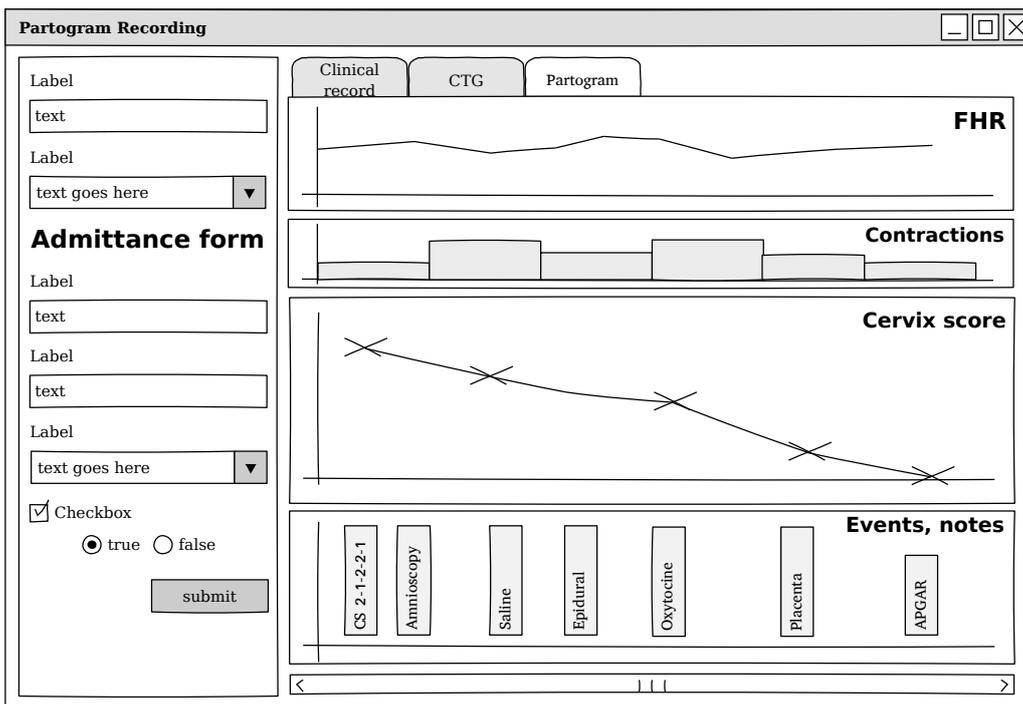


Figure 4.5. Schematic design of electronic partogram UI

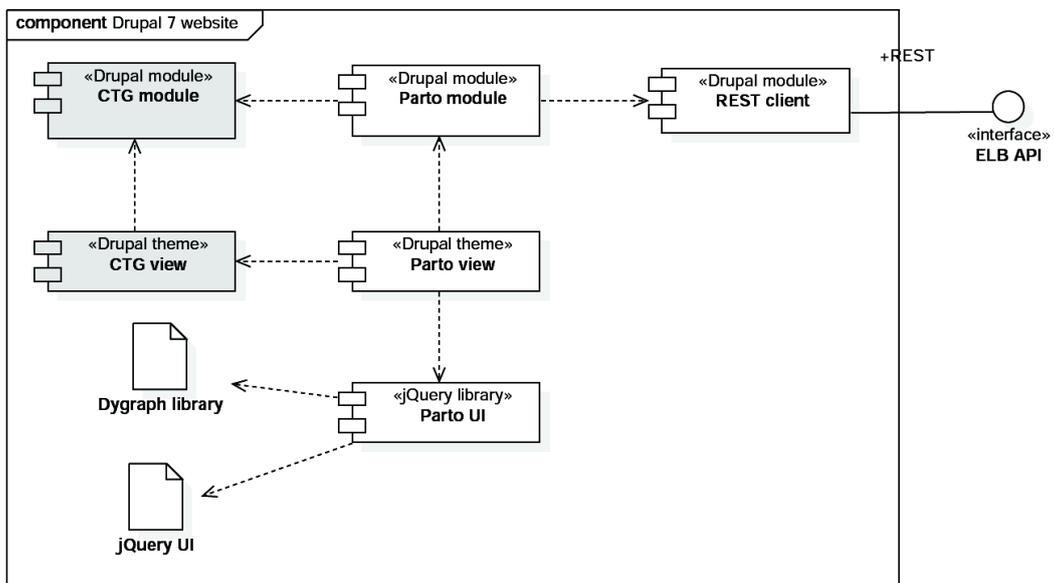


Figure 4.6. Component diagram of CTG and Parto modules integration.

Parto module

This module is a central point of partogram recording website. It depends on two other modules and leverages their functionality into a working solution. Namely, these are the described REST client and the CTG module.

The functionality of the CTG module is necessary, because the whole partogram recording is tied with the original CTG user interface.

Parto module integrates this functionality and implements all the necessary *Drupal hooks*, which are in fact interfaces for interacting with the CMS. These interactions include: menu system integration, template support, JavaScript library assembly, providing web forms, their validation and submission as well as AJAX callback definitions (which are needed for partogram UI).

Parto view

Rendering of partogram UI is performed using *Drupal theme layer*. This layer includes template rendering of various widgets (like menu, buttons, etc.), blocks (like sidebars, login box or forms) or whole pages (including partogram UI page).

There is also a library of components, that are rendered on demand (using AJAX) based on user interaction with the electronic partogram. These interactive elements are closely tied with our JavaScript layer.

Parto UI

The electronic partogram recording involves use of somewhat complicated UI with high level of user interaction, we designed a layer of custom *jQuery plugins* and *JavaScript API* to satisfy this demand.

Since each partogram is basically a chart with annotations, we use the same system for its rendering as the CTG charts do. Therefore we use a JavaScript charting library called *Dygraph* which can render data series using *HTML5 Canvas element*. The library is described in the next chapter, but we should state here, that in order to design an extensible system of partogram charts a wrapper classes of Dygraph with extended functionality need to be implemented.

The decomposition of the UI design is following: for each “signal” there will be a chart with both default and custom settings. The default settings includes point CRUD methods mapped to mouse buttons. The custom settings include chart-specific interactions like context menu, tooltips and annotations, but also a configuration of Dygraph parameters (width, height, axes, colors, etc.). This architecture of our JavaScript charts layer with its fundamental components is shown in Fig. 4.7.

4.7 CBR

With CBR application design we go back to Java language and Spring framework. The CBR application is designed to read data from ELB API, process them into

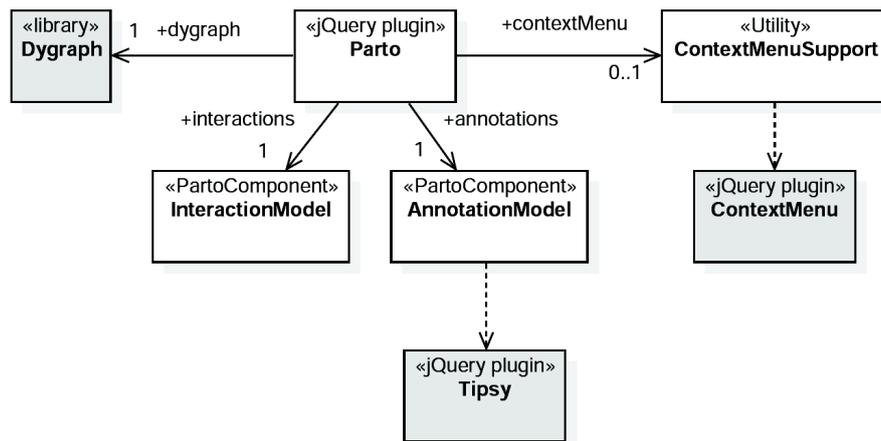


Figure 4.7. Diagram of JavaScript classes and jQuery plugins, third party dependencies are in gray color.

cases and performs experiments with the case base. Since there is no complex user interaction involved, a standard console interface is used.

When started, the application provides commands to insert a similarity search query and to compute and export the whole similarity matrix. In order to do this, the application delegates the tasks to several functional components.

■ 4.7.1 Application structure

Fig. 4.8 shows an overview of the application components architecture. We can see that the design is somewhat similar to ELB API. There is an **API client** with the **DTO package** we described earlier, a **Core component** that wires everything together, a **CBR controller** with logic that performs CBR related operations, a **CBR component** package containing case descriptors and solutions and a package of **similarity definitions**.

The cases are kept in a **linear case base**. The case base is constructed via so-called connector components. Framework **jColibri** provides a **Connector** interface which we implement in our **ElbConnector** class. This connector has a direct access to the REST client and therefore to all the clinical data as well.

Note that, we did not design and implement all the logic and algorithms behind CBR like nearest-neighbor search or similarity evaluation of hierarchies. We only provide a case description, solution, similarity functions, attribute weights and mapping. The rest of the logic and computation is included in the jColibri framework.

Now we will describe the most important topics of the design, the case representation design and similarity mapping.

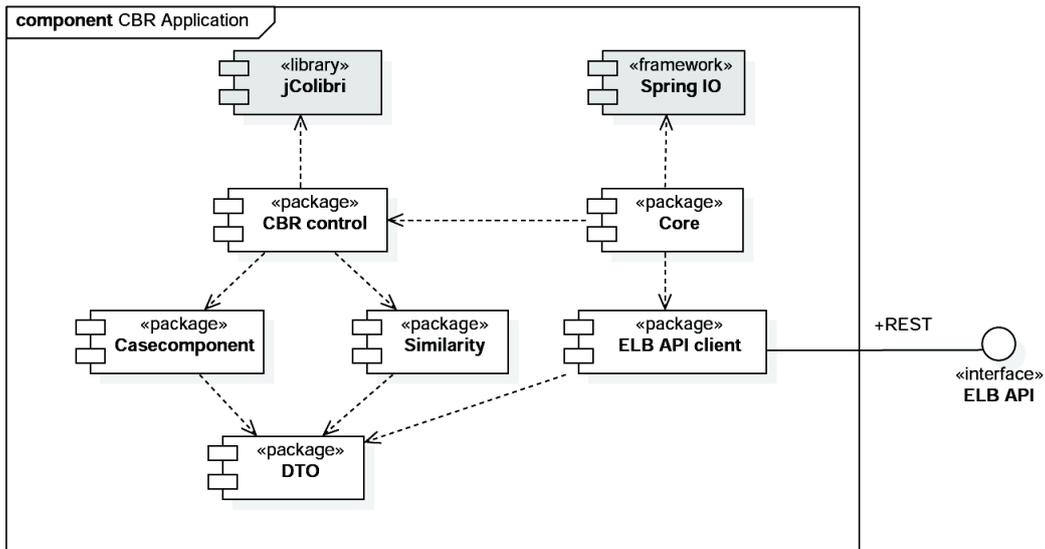


Figure 4.8. Component diagram of CBR application.

4.7.2 Case structure

Here we propose a clinical case representation with link to partogram and cardiotocogram. We start with a diagram of the case representation as a whole in Fig. 4.9. After that we will dive into a deeper level and give a detailed overview of both the case description and solution.

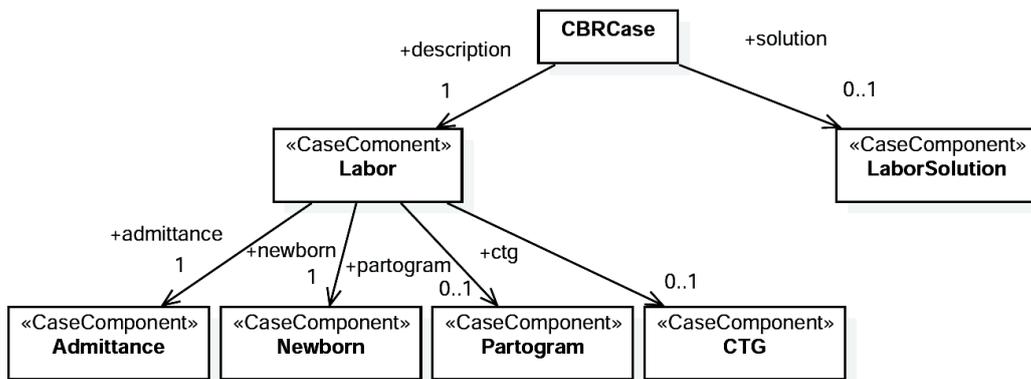


Figure 4.9. Class structure of CBR case representation.

The case structure is **hierarchical** and there is a tight bond with the clinical record structure. Thanks to this bond the mapping between a case and a clinical record in DTOs is quite trivial. This fact has a great benefit in question of the case extensibility – by extending the clinical record we can easily modify the case description (or solution), which is not that easy if a flat case structure is used.

■ 4.7.3 Case description

We shall now describe each *CaseComponent*, starting from the tree root.

Labor

With the hierarchy involved, there are many possible representations of the case description. We have chosen Labor as the root of our case description, because it identifies a particular concept in the data model and because it contains attributes, that we consider the most important for decision support.

These attributes are summarized in Table 4.1, which also gives an information about each attribute type, importance and similarity function used. The weights presented here were chosen according to our understanding of the data and preliminary experiments, but their influence is further examined during Chapter 6 Data analysis.

We also want to examine the influence of *partogram* and *CTG* attributes, since not all the clinical records are complete and contain these two types of data.

attribute	data type	weight	similarity
blood loss	Integer	1.0	QuadraticCutoff
pregnancy days	Integer	1.0	QuadraticCutoff
first stage of labor time	Integer	0.5	Euclidean
induced delivery	Boolean	1.0	Equal
admittance	Admittance	1.0	Average
newborn	Newborn	1.0	Average
partogram	Partogram	1.0	Average
CTG	CTG	1.0	Average

Table 4.1. Attributes of the Labor their mapping to weighted similarities

The similarity functions are described in the next section, so we shall only provide here only a brief explanation of why is a particular similarity used.

There are two main types of similarity functions: simple attribute similarities (Linear, Quadratic, etc.) and complex ones (e.g. Average). Simple functions are used with primitive attribute types, whereas the complex ones with complex types (i.e. an attribute consisting of other sub-attributes). The simple similarities correspond to a distance in the feature space, but the complex ones usually aggregate similarity values of their children in the graph.

Additionally, our simple functions generally differ only in how much importance they give to close attributes and how much to distant ones. As mentioned in Chapter 3, this measure can be linear, polynomial or other metric.

Admittance

We descend one level deeper into the case description tree and provide an overview of Admittance attributes. Each Admittance instance contains attributes whose values are defined for when a mother arrives at the hospital. These attributes are given in Table 4.2.

attribute	data type	weight	similarity
gravity	Enum	0.5	Equal
parity	Enum	1.0	Equal
number of abortions	Integer	1.0	QuadraticCutoff
labor preparation	Boolean	0.1	Equal
mother age	Integer	1.0	QuadraticCutoff
admittance reason	Enum	0.5	Equal

Table 4.2. Attributes of Admittance and their mapping to weighted similarities

It should be noted, that we strictly distinguish the gravity and parity attribute values, instead of comparing them numerically. This means that similarity of these attributes is defined as a relation – the classes are either completely similar or dissimilar.

A different situation is with the number of abortions – this is a virtual attribute computed as the difference between gravity and parity. The value is then assessed using quadratic similarity function.

Newborn

With Newborn, the total of six attributes were included in the case description, these are summarized in Table 4.3.

It may seem confusing, that attributes *position of mother* or *second stage of labor time* belong to the Newborn class. But this is intended, since there may be more children connected to a labor in the data model. Therefore both the *position of mother* and *second stage of labor time* can differ with each child.

Additionally, we consider the attributes *weight* and *sex* as antenatal estimates. Because of this, we will focus on the impact of these two attributes on the CBR results in Chapter 6 Analysis.

attribute	data type	weight	similarity
position of mother	Enum	0.1	Equal
position of fetus	Enum	0.5	Equal
anesthesia	Enum	0.1	Equal
second stage of labor time	Integer	0.1	Euclidean
weight	Integer	0.5	Euclidean
sex	Enum	0.1	Equal

Table 4.3. Attributes of Newborn and their mapping to weighted similarities

CTG

We decided not to evaluate the source CTB signals directly (which could be done using CBR), but instead we want to assess the similarity according to computed *FIGO features* described in Chapter 2.

As shown in Table 4.4, we include FHR baseline, accelerations and decelerations which are all represented as time series, but more attributes could be added. Our method of similarity assessment for time series is described in Section 4.7.5.

attribute	data type	weight	similarity
baseline	Series	1	CosineRCF
accelerations	Series	1	CosineRCF
decelerations	Series	1	CosineRCF

Table 4.4. Attributes of CTG and their mapping to weighted similarities

Partogram

As we mentioned, the partogram data come in the form of *Events*. Since these Events are generally some kind of information defined in time instants, we treat them as time series – i.e. the same way we handle CTG attributes.

However for practical reasons, there are two exceptions. First, the *diagnoses* should probably be considered time series too, but in the original *ELB* data model, they did not contain necessary time resolution. When we started a migration of data to the new model, the mandatory timestamps of each diagnosis were all defined as start of the corresponding childbirth.

With all diagnoses in a case having the same time, we can simply treat them as sets and derive the similarity measure using set intersections (described later).

Second, because usually only few medication events occur during each Labour, we treat them as the sets too.

attribute	data type	weight	similarity
medications	Set	1	JaccardIndex
diagnoses	Set	1	JaccardIndex
cervix dilation	Series	1	CosineRCF
contractions	Series	1	CosineRCF

Table 4.5. Attributes of Partogram and their mapping to weighted similarities

Each Event can be either medication, diagnosis, intervention, observation or measurement. As seen in Table 4.5, we include medication, diagnoses, cervix dilation and contractions (both being observations) to the case description, since these would seem to have the most impact on the delivery result. Our data do not contain enough measurements, so they were left out. The interventions are considered as case solution.

■ 4.7.4 Case solution

The case solution provides an answer to the problem description (query). This answer can be defined in many different ways according to one's perspective and application needs.

For example, we could solve a problem of labor classification and be interested in oxytocine administration during childbirth. Then the query would contain all the relevant data leading to a decision of the drug administration. The answer would be a simple boolean value *administer oxytocine: 1/0*.

In our application we focus on a problem of decision support during childbirth. Therefore we include *interventions* in the solution – *Events* that carry information about procedures performed during labor (e.g. caesarean section, forceps, VEX, hysterectomy, epiziotomy and more).

All these procedures are usually recorded in partogram and can be obtained for each childbirth from the data model.

In Chapter 6 we are going to compare similar cases and clusters they form in the case base. Solutions are not needed for that kind of experiment. However, we are going to analyze each cluster further while taking the solution distribution into consideration.

■ 4.7.5 Similarity functions

In this section we give an overview of similarity functions, that we used in case based reasoning.

As mentioned, there are two main types of similarity functions: those dealing with simple attributes and those dealing with complex ones. We only use one complex function called *Average*, which comes directly from the *jColibri* framework. This function aggregates all the nested similarities of case subcomponents using arithmetic average.

The simple functions for similarity assessment include: *Euclidean*, *Squared Euclidean*, *Squared Euclidean with cutoff*, *CosineRCF* and *JaccardIndex*. Courses of the first three functions are compared in Fig. 4.10. We will now describe each similarity function in detail.

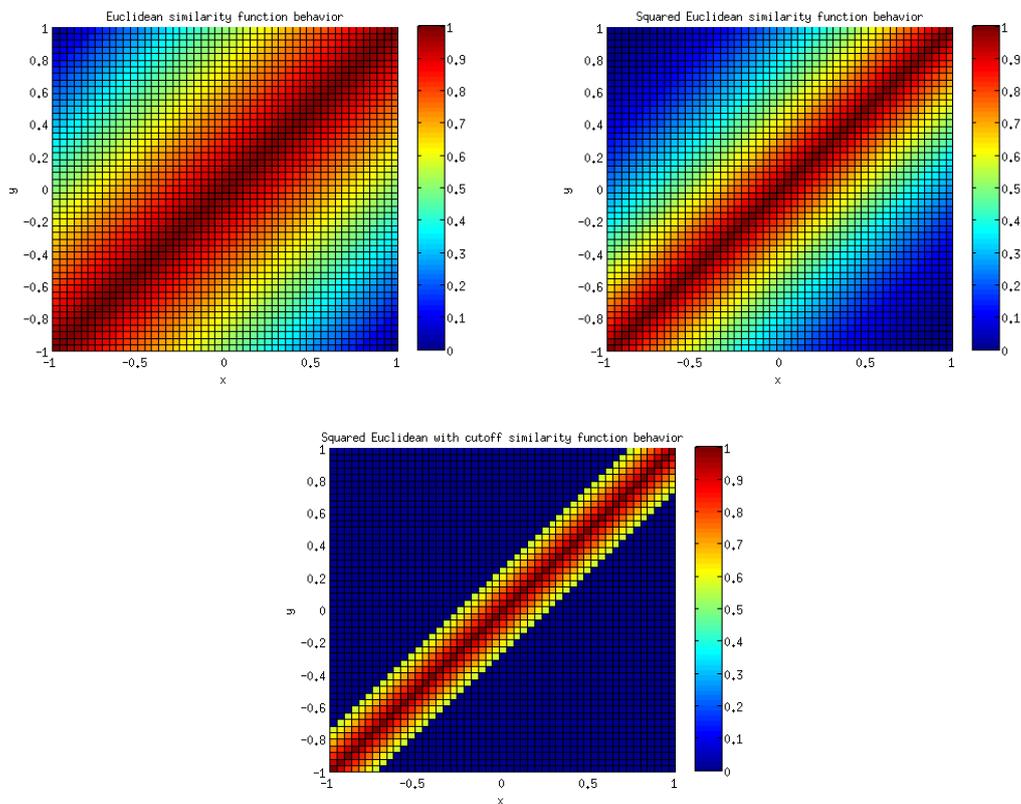


Figure 4.10. Differences in similarity (distance) functions: Euclidean, Squared Euclidean and Squared Euclidean with cutoff $\theta = 0.25$.

Euclidean

This similarity function is based on linear distance measure. Its computation formula is generally very well known and it has already been listed in Chapter 3.4. This function puts linearly increasing importance on similarity of (linearly) closing features.

Squared Euclidean

By squaring the Euclidean similarity measure, we get a similarity that progressively prioritizes closer and penalizes farther points in the feature space..

This has a desired effect in situations where evaluation of attributes differs rapidly even if the actual numeric value shifts slowly (in very fine resolution). Please note that we exponentiate the *similarity*, not the *distance*, otherwise the effect would be opposite.

Squared Euclidean with cutoff

In situations where we know that two relatively close points in feature space are in fact completely dissimilar and we want to put a strong border between them, we use a method of similarity threshold.

We compare two values and if their mutual distance is greater than a threshold θ , we say that their similarity is 0 (we strictly split/cut off these two values). Otherwise the similarity value is given by the underlying function, which in our case is the Squared Euclidean.

A formula summarizing the above is given below:

$$sim(\mathbf{x}, \mathbf{y})_{SECut} = \begin{cases} sim(\mathbf{x}, \mathbf{y})_{SE} & \text{for } \sum_{i=1}^n |x_i - y_i| < \theta, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The impact of the cutoff varies with the given threshold. The difference between Squared Euclidean used with and without the cutoff can be observed in Fig. 4.10.

CosineRCF

We use the **cosine similarity** function (defined in Chapter 3.4) to evaluate the similarity between two time series of *Events* (originating in partogram or CTG data).

The disadvantage of the cosine similarity is addressed in a way presented by Gundersen in [49] – i.e. by multiplication the cosine similarity by so-called **Relative Component Fraction (RCF)**:

$$RCF(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^n \min(x_i, y_i)}{\sum_{i=1}^n \max(x_i, y_i)} \quad (2)$$

$$sim_{CosineRCF}(\mathbf{x}, \mathbf{y}) = RCF(\mathbf{x}, \mathbf{y}) \times sim_{cos}(\mathbf{x}, \mathbf{y}) \quad (3)$$

This operation takes both angle and magnitude of the vectors into consideration and produces a similarity measure. “*This similarity measure will handle both parallel vectors and vectors of equal magnitude pointing in different directions.*” [49]

JaccardIndex

For attributes, that are represented in form of *Sets* (i.e. they either contain a value or not), we compute the similarity using *Jaccard index*, described in Chapter 3.4).

Chapter 5

Implementation

In this chapter we first provide description of technology, frameworks and libraries used in the implemented applications and systems. Next we mention some notable implementation topics for each application separately. Finally we devote a few lines to system testing as well.

5.1 Technology

To implement the designed features we used following third party libraries, frameworks and extensions:

PostgreSQL

“PostgreSQL is a powerful, open source object-relational database system. It has more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness. It runs on all major operating systems.” [57]

The database system is used as the main storage of clinical data, therefore the ELB API is dependent on this platform (using built-in Java connectors).

MySQL

MySQL is another open source database system and perhaps the most popular one. It is used as auxiliary storage for Drupal website (CTG and Parto).

Java

Java is a programming language, which was used to implement ELB. We use the same language to implement ELB API and CBR application. Both these applications are built using Apache Maven software project management tool.

Spring

“Spring IO brings together the core Spring APIs into a cohesive and versioned foundational platform for modern applications. On top of this foundation it also

provides domain-specific runtime environments (DSRs) optimized for selected application types. Spring IO is comprised of the Spring IO Foundation and Spring IO Execution layers. Spring IO is 100 % open source, lean, and modular.” [58]

We use Spring in ELB API and CBR application to scaffold the base application structure and to manage dependencies between components using Inversion of Control (IoC) principle.

Hibernate ORM

“Hibernate ORM enables developers to more easily write applications whose data outlives the application process. As an Object/Relational Mapping (ORM) framework, Hibernate is concerned with data persistence as it applies to relational databases (via JDBC). Hibernate is also an implementation of the Java Persistence API (JPA) specification.” [59]

We use this technology to facilitate data access and mapping between business entities and database relations in ELB API.

Dozer

We include Dozer Java library to JPA API application to manage mapping between business entities and DTOs.

“Dozer supports simple property mapping, complex type mapping, bi-directional mapping, implicit-explicit mapping, as well as recursive mapping. This includes mapping collection attributes that also need mapping at the element level.” [60]

The Dozer mapping is configured using a XML file `dozer-mapping.xml` as shown in following excerpt:

```
<mapping>
  <class-a>cz.cvut.fel.epkapi.dto.NewbornData</class-a>
  <class-b>cz.cvut.fel.epkapi.entity.Newborn</class-b>
  <field>
    <a>sex</a>
    <b>sexIdSex.idSex</b>
  </field>
  <field>
    <a>anesthesia</a>
    <b>anesthesiaIdAnesthesia.idAnesthesia</b>
  </field>
  <field>
    <a>fetalPosition</a>
    <b>fetalPositionIdFetalPosition.idFetalPosition</b>
  </field>
  <!-- etc. -->
</mapping>
```

Electronic Labor Book

As we mentioned in the Introduction, we base our work on the Electronic Labor Book (ELB). ELB is a clinical application with extensive data model which we extend to support partogram recording. *“The user interface is implemented in Java. For storage and retrieval of records PostgreSQL relational database platform is used.”* [2]

Drupal

Drupal is a free, open source web content management system (CMS). Due to its modular design and modular and flexible philosophy it can cover a great number of various use cases.[61] We use as a framework and tool to provide modern and accessible user interface for electronic partogram recording.

Drupal is written mainly in PHP language and requires a web server (Apache or Microsoft IIS), PHP 5.2+ and a database server (MySQL 5.0+, PostgreSQL 8.3+ or SQLite).[61]

Dygraphs

“Dygraphs is a fast, flexible open source JavaScript charting library. It allows users to explore and interpret dense data sets.” [62] It has been used in CTG website to render interactive CTG charts containing millions of data points (see Fig. 5.1) and we further extend its functionality in Parto website UI.

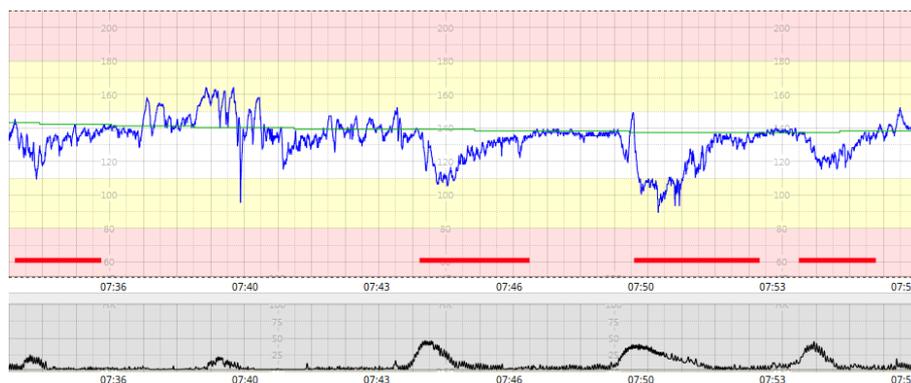


Figure 5.1. CTG signal rendering in HTML5 canvas using Dygraphs

jCOLIBRI

“COLIBRI is a platform for developing academic Case-Based Reasoning (CBR) software. Its main goal is to provide the infrastructure required to develop new CBR systems and its associated software components.” [9] COLIBRI platform

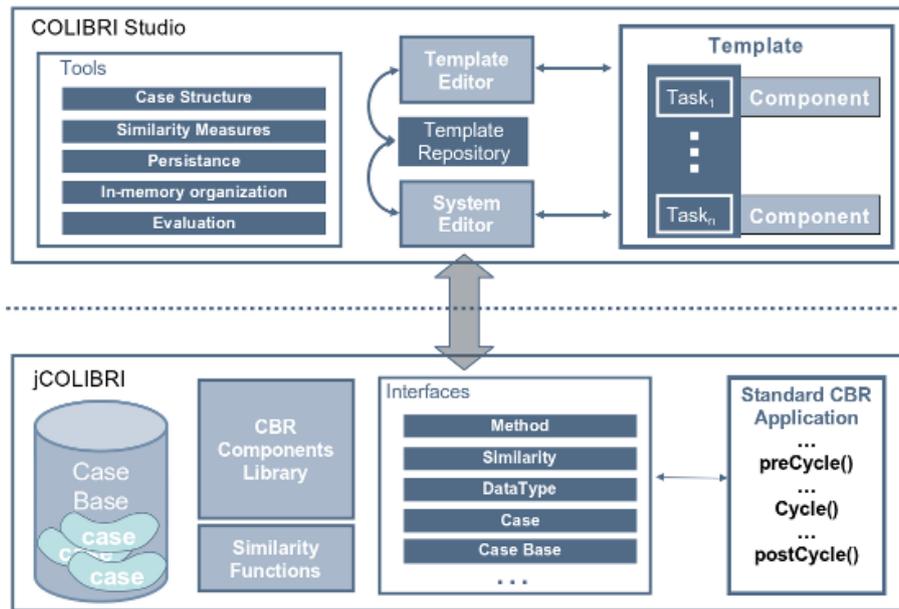


Figure 5.2. The main design principle of COLIBRI is its layered structure, image taken from [9]

consists of two main system layers: jCOLIBRI (a framework for developing CBR applications in Java) and COLIBRI Studio (IDE). [9] This architecture is shown in Fig. 5.2.

5.2 ELB API

As we already mentioned, the ELB API is designed as a web service with REST interfaces. With Spring IO this application can be deployed to a Java web server or it can be run as standalone web server application using Spring Boot.

Once the web application is started a console commands are provided, so one can easily run install procedure (to install some data structure to a clean database) or migrate data from previous ELB database.

With the API running, its endpoints can be accessed at following URLs:

- *GreetingService* – <http://localhost:8080/greeting> (test health status only)
- *FolderService* – <http://localhost:8080/folder>
- *EventService* – <http://localhost:8080/event>
- *EventParameters* – <http://localhost:8080/parameter-template>
- *PersonnelService* – <http://localhost:8080/personnel>

The port that the web server listens on, database connection settings and other properties can be configured in `application.properties` file. The installation guide, documentation and SQL scripts (with the designed database structure and a DBDesigner Fork model) are included within the application CVS repository.

5.3 Parto website

We provide a brief overview of implemented features and screenshots of the electronic partogram recording interface.

Electronic partogram UI

Figure 5.3 shows the final version of user interface for electronic partogram recording. The left sidebar contains forms with anamnesis data and other entries. The charts on the right contain: fetal heart rate, uterine contractions, cervix dilation curve and finally a chart with annotations describing the rest of observations, medications, interventions, diagnoses and measurements.

The first chart with fetal heart rate also includes a green line, that signalizes FHR signal measurement. The corresponding annotation can be clicked on to switch to CTG website UI.



Figure 5.3. Implementation of Parto website UI

We now provide a sequence of figures that illustrate a procedure of inserting partogram Events:

1. By right-clicking on the chart a context menu (*jQuery.contextmenu plugin*) is invoked as seen in Fig. 5.4. The context menu can be invoked on mobile devices using swipe gesture.
2. On event type selection a popup dialog (*jQuery.dialog plugin*) is displayed providing a selection of an event subtype, see Fig. 5.5.

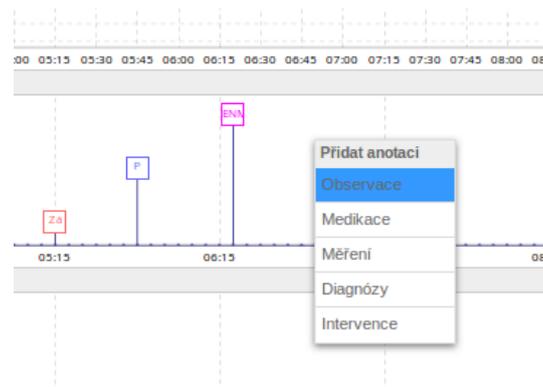


Figure 5.4. Event type selection using a context menu



Figure 5.5. Event subtype selection dialog

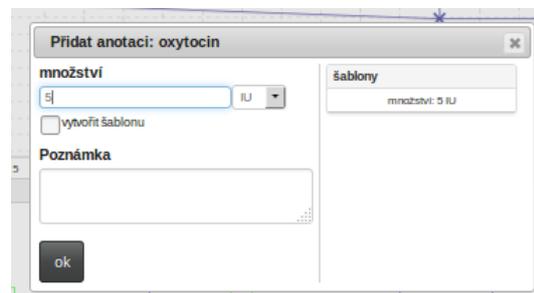


Figure 5.6. Event attribute-value templates on the right (oxytocine dose)

- Once the event subtype is selected, the dialog is automatically submitted. If the event subtype defines any attributes or parameters, another dialog is invoked. During this step, we can choose what attribute templates should be created or we can pick an existing attribute template, see Fig. 5.6.
- If there are no parameters, an event is created at once (step 4).
- If the parametrization is completed and user submits the dialog, a new event is created and rendered using Dygraphs annotations on the corresponding position in time. Immediately after that, a tooltip (*jQuery.tipsy plugin*) is initialized for the new annotation element as seen in Fig. 5.7

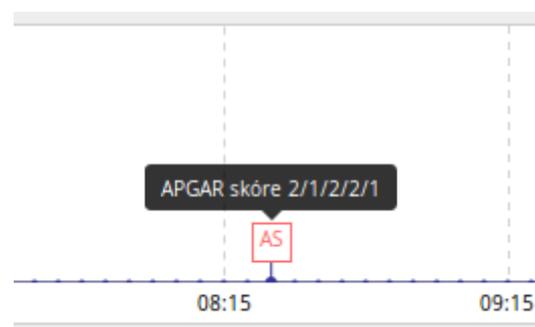


Figure 5.7. New event is created with the selected type and subtype, a tooltip gets initialized.

5. Finally, each parametric event (annotation) can be clicked on, which results in opening the dialog with attribute settings.

5.4 CBR application

Since the CBR application also takes advantage of Spring framework and because it is connected to the API using REST client, its implementation is partially mirrored – this applies for DTOs and web service interfaces. There are also similar console commands for CBR application control and the same configuration file. Other components including CBR CaseComponents, Similarities and mapping of DTOs to CaseComponents are unique.

Following main features are implemented:

- Data loading from ELB API, the amount of cases fetched from the API is controlled in the config file.
- Search function that builds a search query and perform a search through the case base. The procedure returns 5 nearest neighbors – 5 most similar cases.
- Similarity matrix computation where similarity of each two cases is evaluated (only a triangular matrix is computed, since the relation is symmetric and the graph undirected).
- Labeling using case solutions to support cluster analysis.

5.5 Testing and results

We conclude this chapter with a brief summary of testing, that we performed with the resulting system.

We consider the first stage of testing the so-called *Unit tests*, that cover small chunks of application logic and can be evaluated automatically and repeatedly. This way we test REST services of ELB API, CBR REST client, CBR DTO to case description mapping and similarity functions.

The other stage of testing is covered with user tests evaluation the application. This was performed as well since the paper partograms needed to be manually inserted into electronic form. During this testing a few issues appeared.

First it became obvious that time windows for labors in the ELB database and paper partograms do not correspond very well (i.e. the labors often start hours

before the partogram is recorded). This resulted in very large electronic partograms with initial gaps in charts. This issue could be resolved with some sort of partogram time scale and time bounds adjustment feature.

Second, data traffic between the Parto website and ELB API during partogram recording is larger than we anticipated and sometimes results in lags. This issue was resolved using API results caching in Drupal.

The last stage of testing is performed in the next chapter, where we experiment with various similarity matrices and study clusters of similar cases.

Chapter 6

Data analysis

In this chapter we focus on analysis of clusters formed in similarity matrices created with our *CBR application*. We start with an introduction about what data is available, then we introduce tools and methods of the experiments and data analysis and finally we add discussion over the results.

6.1 Introduction

We provide an introduction of what data we can deal with during our similarity assessment experiments. In the *ELB* database there are 5476 labors, **5348 of which are with a single newborn**. As we reasoned earlier, these labors are of our interest. There are 5346 mothers connected with these labors. There are **2311 different diagnoses, 8 medications and 17 interventions** in total.

If we look at partogram data, there are 21797 events in total, 13497 of them are connected with diagnoses, 596 with medications and 6238 with interventions, some events come from manually inserted electronic partograms. This makes it almost **3 diagnoses and 1 intervention per labor and 1 medication per 9 labors** on average.

There are 57 labors with CTG signal data available, however only 9 of them have all necessary FIGO features computed. Also, as only 33 paper partogram copies corresponding to the *ELB* database are available to us, the impact of Partogram case component can not be identified reliably. For these reasons we will be making several experiments in the next section, each with different setup of case components and case attributes.

6.2 Methodology

We present our methodology for experiments with CBR similarities. Each experiment consists of following steps:

- 1) Define similarity mapping.
- 2) Load random collection of 1000 i.i.d. cases from EPK API.

- 3) Compute a similarity matrix.
- 4) Export the matrix to CSV with nodes labeled using case solutions.
- 5) Import the CSV file to Gephi graph visualisation tool.
- 6) Filter graph edges and orphan nodes.
- 7) Compute graph (network) modularity. If the modularity is too low, adjust attribute weights and try again.
- 8) Color the graph using the discovered modularity classes.
- 9) Lay out the graph using OpenOrd algorithm.
- 10) Observe clusters in the similarity graph and try to find coherence between the cases and their solutions in each cluster.

■ 6.2.1 Encoding of attributes

In our experiments we encode enumerations of some attributes to save space and performance. These attributes are: interventions (solutions), admittance reason, mother position, fetal position, sex and anesthesia. Additionally, we encode multiple interventions using the same code – that’s because we want to minimize the total number of possible solutions and these interventions are considered very similar in our application. A key to this enumeration coding is given in Attachment C.

The vertices of the similarity graph are labeled using “SOLUTION_DESCRIPTION” pattern, therefore it is possible to connect each node with the original case description (labor) and solution (interventions).

■ 6.2.2 Reduction of diagnoses

Since there is a great number of diagnoses, we decided to group them using their relations and further work with diagnose groups, instead of the individual diagnoses. Because we use *ICD-10* classification codes¹⁾, which is a hierarchical taxonomy, the grouping can be performed by taking into account only the first three characters of each diagnose code. With this technique we obtain a total count of 57 diagnosis groups only.

■ 6.2.3 Tools

Before we begin with the experiments description, we shall briefly describe the tools we use – namely Gephi, network modularity and OpenOrd.

Gephi

Gephi is open-source graph visualization and manipulation tool and Java library. It allows real-time visualization of large graphs (consisting of thousands of nodes

¹⁾ International Classification of Diseases (ICD), <http://www.who.int/classifications/icd>

and edges), it provides algorithms for community detection (modularity) and graph-topology layout algorithms – some are based on force, but there are also ones that work with edge weights (similarities) like OpenOrd. [63]

We use this tool to visualize the CBR similarities coming in a form of similarity matrix. We assume, that usage of this tool does not alter the results. The visualization consists of two tasks only:

1. Coloring the similarity matrix using network modularity analysis (see Fig. 6.2)
2. Creating a layout of the corresponding similarity graph using OpenOrd algorithm (see Fig. 6.1)

With the layout computed, we can switch to a view of the clusters where each case is labeled using its IDs (see Fig.6.3).

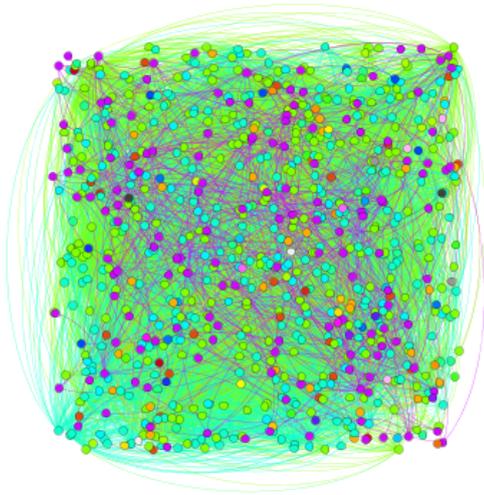


Figure 6.1. Similarity graph coloring using graph modularity.

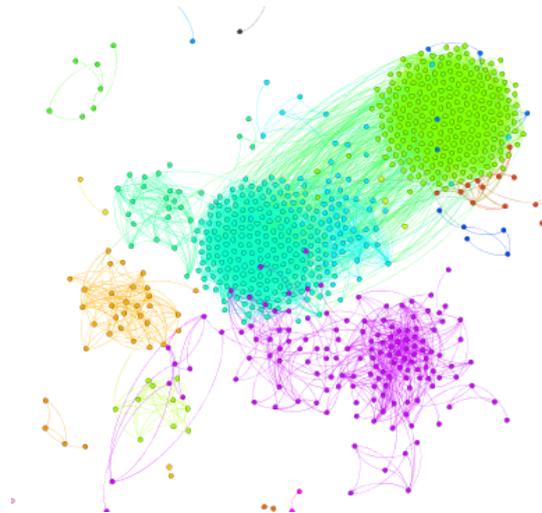


Figure 6.2. Similarity graph layout using OpenOrd algorithm.

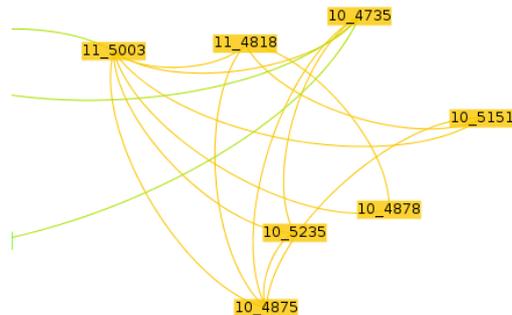


Figure 6.3. Labeling of cases in Gephi.

Network modularity

Gephi uses a heuristic algorithm proposed by Blondel et. al. [64], that can extract community structures of large graphs using modularity optimization technique. We can benefit from this community (or class) discovery in our visualization if we color each community in different color. The modularity of a network is one of its structural properties and since the structure of our graph is given by similarity, we can infer that the graph coloring relates closely to the similarity among cases in the computed base.

OpenOrd algorithm

To layout the similarity graph we use OpenOrd algorithm implementation in a Gephi plugin [65].

“OpenOrd is a graph drawing algorithm specialized for drawing large-scale real-world graphs. The algorithm uses edge-cutting, average-link clustering, multilevel graph coarsening, and a parallel implementation of a force-directed method based on simulated annealing.” [66]

The algorithm can handle very large graphs (over 1 million nodes) and works with weighted undirected graphs and provides good overall results on the data we are dealing with.

6.3 Experiments

In this section we want to study influence of different attributes and weights on the case clusters and their relation to the case solutions (interventions).

For this purpose we perform four experiments, each with different number of attributes included in CBR evaluation. We are trying to find a good visualization of the similarity matrix, therefore we repeat each experiment several times with different lower bound of similarity. Sometimes we find more than one configuration of interest.

Each experiment description consists of three parts: setup overview, analysis of major clusters and evaluation. During cluster analysis we compute the ratio of individual solutions inside each cluster. We also provide three random samples from each major cluster for the reader in Attachment D.

These are the individual experiment setups:

1. The first experiment includes only the basic clinical data (i.e. Labor, Admittance and Newborn) in its case description. Weight and sex attributes (anteperatal estimates) are left out.

2. The second experiment is the same as the first one, except it includes these estimates.
3. The third experiment adds partogram case component with diagnoses only.
4. The last experiment adds CTG and partogram with all the attributes defined in Chapter 4.7.

■ 6.3.1 Experiment 1

The experiment setup is following: the similarity is computed using clinical data without the antepartal estimates. The similarity lower bound is 0.85.

The similarity graph contains a total number of 483449 edges. By filtering edge weights (similarities) lower than 0.85, this number is reduced to 59035 (12.21 %). There are 892 nodes, the average node degree is 132.4 and the network modularity 0.397 (15 total classes). An overview of graph clusters (communities) is given in Fig. 6.4.

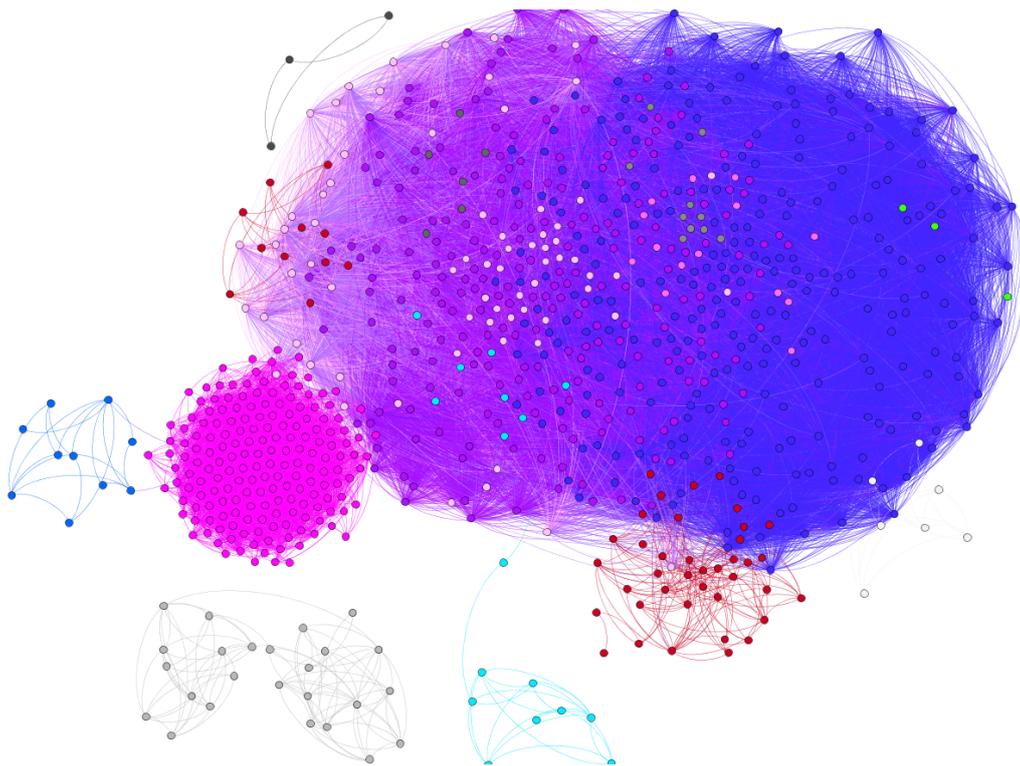


Figure 6.4. Experiment 1 - similarity clusters overview (clinical data, $sim_{min} = 0.85$).

Cluster analysis

cluster	case solution	solution ratio
4. (blue)	10 (episiotomy)	137 (51.1 %)
4. (blue)	11 (rupture)	61 (22.8 %)
4. (blue)	12 (labor sine)	40 (14.9 %)
4. (blue)	1 (acute caesarean section)	17 (6.3 %)
5. (violet)	12 (labor sine)	81 (36.7 %)
5. (violet)	11 (rupture)	62 (28.1 %)
5. (violet)	10 (episiotomy)	62 (28.1 %)
6. (pink)	10 (episiotomy)	73 (41.7 %)
6. (pink)	11 (rupture)	42 (24 %)
6. (pink)	12 (labor sine)	34 (19.4 %)
6. (pink)	1 (acute caesarean section)	16 (9.1 %)
8. (light pink)	12 (labor sine)	48 (68.6 %)
8. (light pink)	11 (rupture)	16 (22.9 %)
8. (light pink)	10 (episiotomy)	4 (5.7 %)
0. (red)	2 (primary caesarean section)	29 (63 %)
0. (red)	1 (acute caesarean section)	15 (32.6 %)
10. (gray)	1 (acute caesarean section)	22 (91.7 %)
10. (gray)	2 (primary caesarean section)	2 (8.3 %)

Table 6.1. Analysis of similarity clusters in relation to case solutions

Evaluation

The clusters are quite compact when looking at the case descriptions (D). We can clearly see common features inside each cluster like gravidity and parity in clusters 4, 5 and 6; anesthesia in cluster 5; zero first stage times in the third cluster or induced labors in the fourth cluster.

However, when looking at case solutions, their intra-cluster variability seems to be very high with this setup. Not only there are many solutions inside clusters, but the solutions are quite different as well (including labor sine, caesarean, ruptures and others). The only exception seems to be cluster 10. This small cluster has 91.7 % of acute caesareans and the rest are primary caesareans. Majority of these cases are labors with newborn in asynclitism.

6.3.2 Experiment 2a

The experiment setup is following: the similarity is computed using clinical data with antepartal estimates. The similarity lower bound 0.90.

There are 791 (79.1 %) nodes and 13901 (2.86 %) edges in the graph. The average node degree is 35.2 and network modularity is 0.566 (32 classes). The communities are visualized in Fig. 6.5.

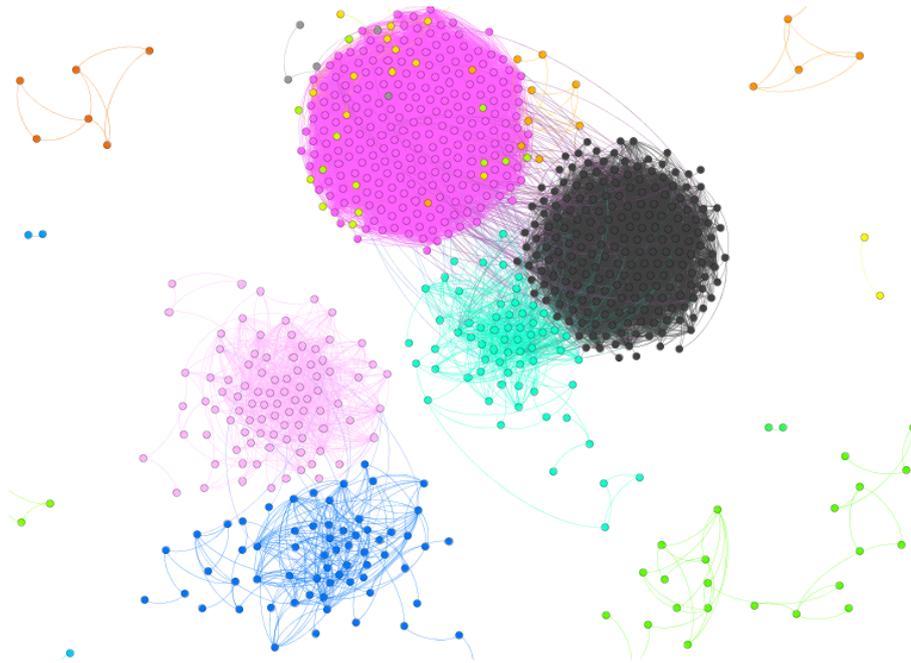


Figure 6.5. Experiment 2a – similarity clusters overview (clinical data, weight and sex, $sim_{min} = 0.90$).

Cluster analysis

cluster	case solution	solution ratio
0. (magenta)	10 (episiotomy)	127 (55.5 %)
0. (magenta)	11 (rupture)	55 (24 %)
0. (magenta)	12 (labor sine)	33 (14.4 %)
20. (black)	12 (labor sine)	72 (36.5 %)
20. (black)	11 (rupture)	59 (29.9 %)
20. (black)	10 (episiotomy)	56 (28.4 %)
24. (cyan)	10 (episiotomy)	48 (62.3 %)
24. (cyan)	11 (rupture)	15 (19.5 %)
24. (cyan)	12 (labor sine)	7 (9.1 %)
24. (cyan)	1 (acute caesarean section)	4 (5.2 %)
21. (pink)	12 (labor sine)	40 (70.2 %)
21. (pink)	11 (rupture)	14 (24.6 %)
21. (pink)	10 (episiotomy)	3 (5.3 %)
19. (blue)	10 (episiotomy)	18 (37.5 %)
19. (blue)	11 (rupture)	18 (37.5 %)
19. (blue)	12 (labor sine)	8 (16.7 %)
19. (blue)	7 (VEX)	4 (8.3 %)

Table 6.2. Analysis of similarity clusters in relation to case solutions

Evaluation

We can find some quite similar cases inside some clusters – their similarity is given by their descriptions, but in sometimes even same solutions can be observed, this applies mainly for smaller clusters (42, 21). Again, intra-cluster variance of gravidity and parity is very low in almost each cluster (notice primiparas in 0, 24 and 21 in contrast to multiparas in 20 and 19). Cluster 3 also includes a higher ratio of abortions.

By looking at the antepartal estimates, we can observe, that some clusters tend to be formed around same sex (clusters 20, 19, 24, 21). However distribution of fetal weights seems not to correspond tightly with any of the clusters. Perhaps further experiments with different fetal weight similarity functions could be performed.

6.3.3 Experiment 2b

The experiment setup is following: the similarity is computed using clinical data, antepartal estimates. The similarity lower bound is raised to 0.95.

There are 211 (21.1 %) nodes and 905 (0.19 %) edges in the graph. The average node degree is 8.6 and network modularity is 0.641 (13 classes). The communities are visualized in Fig. 6.6.

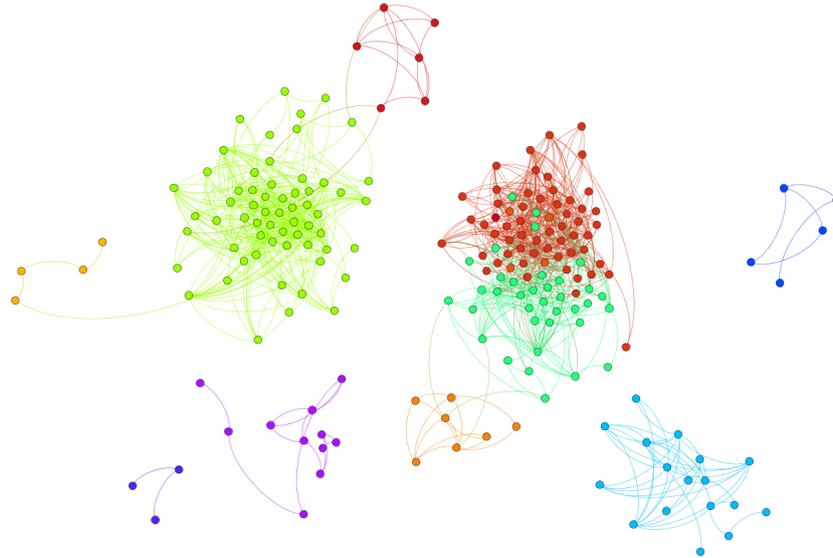


Figure 6.6. Experiment 2b – similarity clusters overview (clinical data, weight and sex, $sim_{min} = 0.95$).

Cluster analysis

cluster	case solution	solution ratio
0. (green)	10 (episiotomy)	66 (51.6 %)
0. (green)	11 (rupture)	31 (24.2 %)
0. (green)	12 (labor sine)	19 (14.8 %)
0. (green)	1 (acute caesarean section)	9 (7 %)
20. (red)	12 (labor sine)	44 (35.8 %)
20. (red)	11 (rupture)	38 (30.9 %)
20. (red)	10 (episiotomy)	31 (25.2 %)
24. (cyan)	10 (episiotomy)	40 (61.5 %)
24. (cyan)	11 (rupture)	12 (18.5 %)
24. (cyan)	12 (labor sine)	7 (10.8 %)
7. (blue)	10 (episiotomy)	25 (43.1 %)
7. (blue)	12 (labor sine)	20 (34.5 %)
7. (blue)	11 (rupture)	13 (22.4 %)
2. (violet)	10 (episiotomy)	32 (64 %)
2. (violet)	11 (rupture)	13 (26 %)
2. (violet)	12 (labor sine)	5 (10 %)
19. (orange)	11 (rupture)	17 (39.5 %)
19. (orange)	10 (episiotomy)	15 (34.9 %)
19. (orange)	12 (labor sine)	7 (16.3 %)
19. (orange)	7 (VEX)	4 (9.3 %)

Table 6.3. Analysis of similarity clusters in relation to case solutions

Evaluation

As we increased the lower bound of similarity, we reduced the total number of edges (and orphan nodes). This should theoretically result in formation of clusters with very low intra-cluster variance of cases. We can observe that the clusters are formed around gravidity/parity combinations, pregnancy days and also newborn weights. The first cluster seems to contain cases with longer first stages of labor (around 300 minutes), while the third cluster contains majority of cases with the first stage lesser than 200 minutes.

The variance of solutions inside clusters is relatively high, but most of the clusters are formed around cases with related solutions (episiotomy and ruptures).

6.3.4 Experiment 3a

The experiment setup is following: the similarity is computed using clinical data, antepartal estimates and partograms with diagnoses and medications only. The similarity lower bound is 0.85.

There are 711 (71.1 %) nodes and 18096 (3.74 %) edges in the graph. The average node degree is 50.9 and network modularity is 0.444 (11 classes). The communities are visualized in Fig. 6.7.

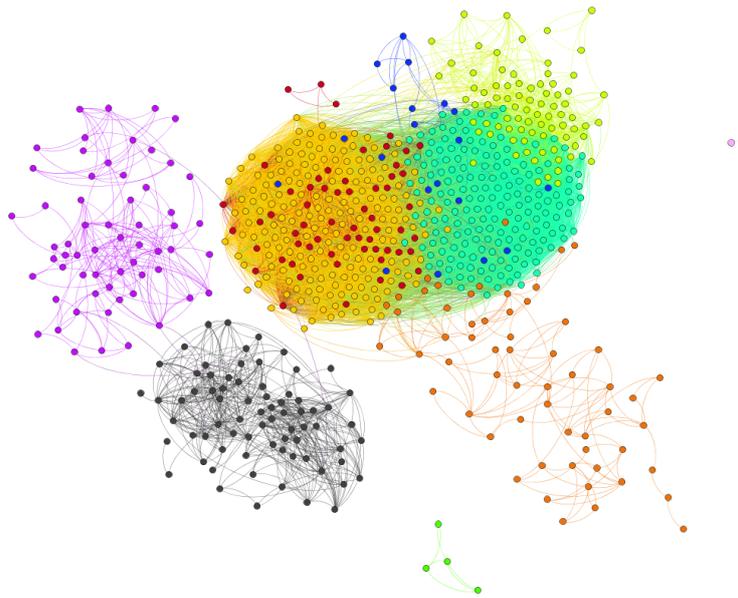


Figure 6.7. Experiment 3a – similarity clusters overview (clinical data, estimates, diagnoses, medications, $sim_{min} = 0.85$).

Cluster analysis

cluster	case solution	solution ratio
0.	1 (acute caesarean section)	109 (31.3 %)
0.	12 (labor sine)	98 (28.2 %)
0.	10 (episiotomy)	53 (15.2 %)
0.	11 (rupture)	43 (12.4 %)
0.	2 (primary caesarean section)	27 (7.8 %)
3.	12 (labor sine)	71 (39.9 %)
3.	10 (episiotomy)	53 (29.8 %)
3.	11 (rupture)	51 (28.7 %)
8.	10 (episiotomy)	92 (54.4 %)
8.	11 (rupture)	47 (27.8 %)
8.	12 (labor sine)	25 (14.8 %)
6.	10 (episiotomy)	45 (60 %)
6.	11 (rupture)	17 (22.7 %)
6.	12 (labor sine)	10 (13.3 %)
10.	10 (episiotomy)	30 (44.1 %)
10.	11 (rupture)	22 (32.4 %)
10.	12 (labor sine)	10 (14.7 %)
4.	10 (episiotomy)	34 (68 %)
4.	11 (rupture)	8 (16 %)

Table 6.4. Analysis of similarity clusters in relation to case solutions

Evaluation

As we include diagnoses and medications into similarity assessment, we can study their distribution across the case clusters and their impact on intra-class variability of solutions.

In this setup we can find the same diagnoses (or more precisely groups of diagnoses) in clusters:

- cluster 0 – diagnosis O32 (maternal care for malpresentation of fetus)
- cluster 3 – diagnosis O24 (Diabetes mellitus in pregnancy)
- clusters 3 and 8 – diagnosis O80 (full-term uncomplicated delivery)

Note that the majority of cluster 0 also ends with caesarean section (which is a common solution for malpresentation as mentioned in Chapter 2). Diagnosis O80 is very common in clusters 3 and 8, however this diagnosis is probably in consent only with cluster 3.

6.3.5 Experiment 3b

The experiment setup is following: the similarity is computed using clinical data, antepartal estimates and partograms with diagnoses and medications only. The similarity lower bound is 0.90.

There are 467 (46.7 %) nodes and 2719 (0.56 %) edges in the graph. The average node degree is 11.7 and network modularity is 0.581 (40 classes). The communities are visualized in Fig. 6.8.

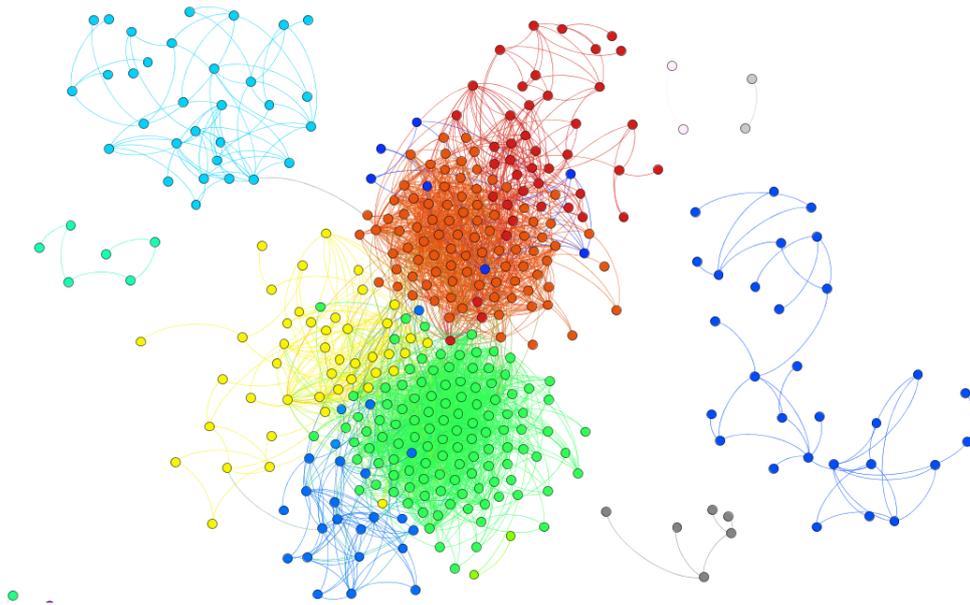


Figure 6.8. Experiment 3b – similarity clusters overview (clinical data, estimates, diagnoses, medications, $sim_{min} = 0.90$).

Cluster analysis

cluster	case solution	solution ratio
0. (red)	1 (acute caesarean section)	106 (35.8 %)
0. (red)	12 (labor sine)	64 (21.6 %)
0. (red)	10 (episiotomy)	51 (17.2 %)
0. (red)	11 (rupture)	31 (10.5 %)
0. (red)	2 (primary caesarean section)	26 (8.8 %)
30. (green)	12 (labor sine)	44 (42.7 %)
30. (green)	11 (rupture)	32 (31.1 %)
30. (green)	10 (episiotomy)	27 (26.2 %)
27. (yellow)	10 (episiotomy)	56 (65.1 %)
27. (yellow)	11 (rupture)	17 (19.8 %)
27. (yellow)	12 (labor sine)	12 (14 %)
4. (blue)	10 (episiotomy)	50 (67.6 %)
4. (blue)	11 (rupture)	15 (20.3 %)
4. (blue)	12 (labor sine)	5 (6.8 %)
6. (azure)	10 (episiotomy)	26 (52 %)
6. (azure)	11 (rupture)	14 (28 %)
6. (azure)	12 (labor sine)	8 (16 %)

Table 6.5. Analysis of similarity clusters in relation to case solutions

Evaluation

We repeat the experiment, but we raise the similarity lower bound to 0.90 and observe the formed clusters. These are again smaller since a lot of edges and nodes are left out. Solutions with high ratio can be found in most of the clusters now (30, 27, 4, 6).

Interestingly, aside from the regular clusters that we observed in the previous experiment, a large number of smaller, but very compact (highly similar – descriptions and solutions) communities were also formed. These smaller communities are completely separated components consisting of few connected cases. These components can be observed in Fig. 6.9, and could be considered characteristic candidates in a decision support system.

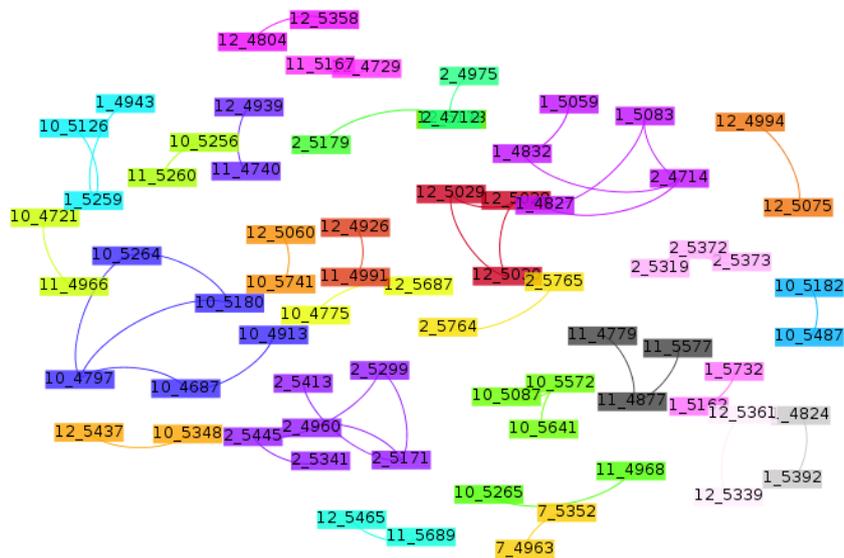


Figure 6.9. Small separated components of the graph with high similarity and low intra-class solution variability.

6.3.6 Experiment 4

The experiment setup is following: the similarity is computed using complete case descriptions – that is clinical data, antepartal estimates, full partograms and CTG with FIGO features. The similarity lower bound is 0.75 (the maximum is 0.82).

There are 323 (32.3 %) nodes and 2009 (0.42 %) edges in the graph. The average node degree is 12.4 and network modularity is 0.587 (18 classes). The communities are visualized in Fig. 6.10.

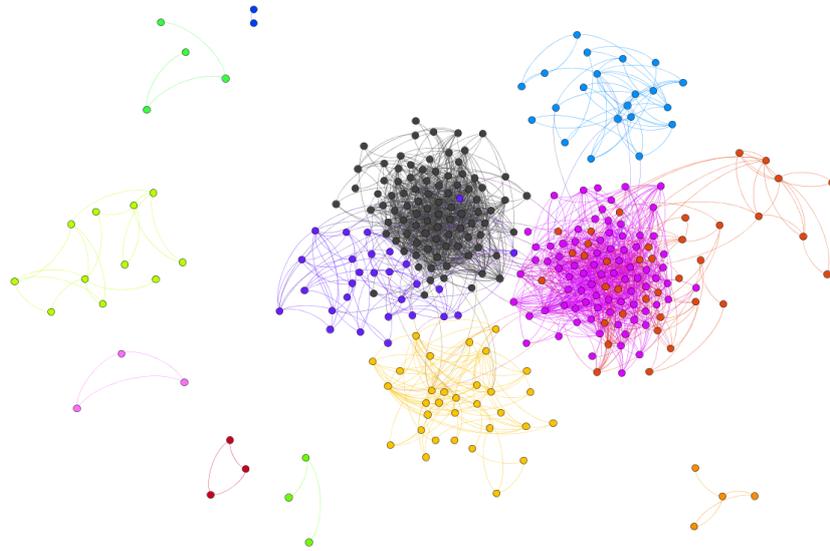


Figure 6.10. Experiment 4 – similarity clusters overview (complete case description, $sim_{min} = 0.75$).

Cluster analysis

cluster	case solution	solution ratio
16. (black)	12 (labor sine)	38 (44.7 %)
16. (black)	11 (rupture)	26 (30.6 %)
16. (black)	10 (episiotomy)	21 (24.7 %)
13. (violet)	10 (episiotomy)	45 (61.6 %)
13. (violet)	11 (rupture)	17 (23.3 %)
13. (violet)	12 (labor sine)	10 (13.7 %)
1. (red)	10 (episiotomy)	19 (65.5 %)
1. (red)	11 (rupture)	7 (24.1 %)
1. (red)	12 (labor sine)	3 (10.3 %)
12. (blue)	12 (labor sine)	11 (40.7 %)
12. (blue)	10 (episiotomy)	9 (33.3 %)
12. (blue)	11 (rupture)	7 (25.9 %)
3. (yellow)	12 (labor sine)	21 (80.8 %)
3. (yellow)	11 (rupture)	5 (19.2 %)
10. (light blue)	10 (episiotomy)	11 (64.7 %)
10. (light blue)	12 (labor sine)	4 (23.5 %)
10. (light blue)	11 (rupture)	2 (11.8 %)

Table 6.6. Analysis of similarity clusters in relation to case solutions

Evaluation

The inclusion of empty partograms and CTG data resulted in more random distribution of the cases, since the overall similarity dropped to a maximum of 0.82. The solution distribution has exceptions as the clusters get smaller (3 and 10). Because the ratio of the cases with full partograms is low, we can not observe their impact on the clusters. The future work should focus on additional data acquisition or perhaps on similarity assessment of partograms alone.



Chapter 7

Conclusion

In this work we have studied the obstetrics theory minimum necessary to cover the issues of electronic partogram and cardiotocogram recording. We performed an update to a clinical database, successfully extending it with data structures for partogram recording in a form of event entities.

We performed design and implementation of a modular system working with this clinical database and allowing partogram recording through a modern rich user interface.

We also devoted a large part of this work to the topic of case-based reasoning – we designed and implemented clinical case description in Java language. We performed testing and resulting data analysis to examine if the similarity among cases leads to meaningful clusters formation.

With this in mind, we conclude that we have covered the all the requirements stated in Chapter 1. Furthermore, the resulting system is build on open-source technologies only.

Analysis of data clusters of the most similar cases showed, that intra-cluster similarity given by case-descriptions can result in low intra-cluster solution variability. As mentioned, the future research should include additional data acquisition, processing and analysis.



References

- [1] HealthIT.gov. *What are the advantages of electronic health records*. 2014. Visited on 2015-05-10.
<http://www.healthit.gov/providers-professionals/faqs/what-are-advantages-electronic-health-records>.
- [2] M. Huptych, V. Chudáček, P. Janků, and L. Lhotská. *An application for obstetrics and neonatology: Electronic labor book*. In: *IFMBE Proceedings*. Beijing: Springer, 2013. 1257–1259. ISBN 9783642293047.
- [3] H. Underwood. *PartoPen: Enhancing the Partograph with Digital Pen Technology*. In: *CHI '12 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2012. 1393–1398. ISBN 978-1-4503-1016-1.
<http://doi.acm.org/10.1145/2212776.2212461>.
- [4] Department of Biomedical Engineering John HopkinsUniversity. *ePartogram*. 2011. Visited on 2015-05-10.
<http://eng.jhu.edu/wse/cbid/page/epartogram>.
- [5] *ApparentApps Partogram*. Visited on 2015-05-08.
<http://apparentapps.com/partogram/site/home/about>.
- [6] S. Bogaerts. *IUCBRF*. 2013. Visited on 2015-05-07.
<http://www.cs.indiana.edu/~leake/iucbrf/index.html>.
- [7] O. Dantas. *jCaBaRe - Case Based Reasoning for Java*. 2013. Visited on 2015-05-07.
<http://sourceforge.net/projects/jcabare/>.
- [8] *FreeCBR*. Visited on 2015-05-07.
<http://freecbr.sourceforge.net/>.
- [9] Group of Artificial IntelligenceApplications. *COLIBRI*. 2015. Visited on 2015-05-10.
<http://gaia.fdi.ucm.es/research/colibri>.
- [10] A. Kvasničková. *The pregnancy and delivery of the first child in higher age of woman, Bachelor thesis*. Masarykova univerzita. 2009.
https://is.muni.cz/th/214739/1f_b/Bak._prace_2009.pdf.

- [11] E. A. Friedman. Primigravid labor; a graphicostatistical analysis.. *Obstetrics and gynecology*. 1955, 6 (6), 567–589. DOI 10.1097/00006250-195512000-00001.
- [12] H. R. Philpott. Graphic records in labour. *British Medical Journal*. 1972, 4 163.
- [13] T. Lavender, A. Hart, and R. M. D. Smyth. Effect of partogram use on outcomes for women in spontaneous labour at term. *Cochrane Database of Systematic Review 2013*. 2013, (7),
- [14] R.S.V. Cartmill, and J.G. Thornton. Effect of presentation of partogram information on obstetric decision-making. *The Lancet*. 1992, 339 (8808), 1520–1522. DOI 10.1016/0140-6736(92)91275-D.
- [15] J. Studd. Partograms and nomograms of cervical dilatation in management of primigravid labour.. *British medical journal*. 1973, 4 (5890), 451–455. DOI 10.1136/bmj.4.5890.451.
- [16] G. Bosse, S. Massawe, and A. Jahn. The partograph in daily practice: it’s quality that matters. *International Journal of Gynecology & Obstetrics*. 2002, 77 (3), 243–244. DOI 10.1016/S0020-7292(02)00004-8.
- [17] World Health Organization. *The partograph. A managerial tool for the prevention of prolonged labour*. 1989.
<http://whqlibdoc.who.int/hq/1988/WHO.MCH.88.3.pdf>.
- [18] World Health Organization. World Health Organization partograph in management of labour. *The Lancet*. 1994, 343 (8910), 1399–1404. DOI 10.1016/S0140-6736(94)92528-3.
- [19] Iffat Javed, Shereen Bhutta, and Tabassum Shoaib. Role of partogram in preventing prolonged labour. *Journal of the Pakistan Medical Association*. 2007, 57 (8), 408–411.
- [20] A. SIZER. A second-stage partogram. *Obstetrics & Gynecology*. 2000, 96 (5), 678–683. DOI 10.1016/S0029-7844(00)00981-9.
- [21] MAXDORF PUBLISHING. *Cervix skóre, Velký lékařský slovník*. 2008. Visited on 2015-04-25.
<http://lekarske.slovniky.cz/lexikon-pojem/cervix-skore-zkr-cs>.
- [22] A. Zwinger, J. Jelínek, P. Ventruba, and V. Unzeitig. *Porodnictví*. Praha: Galén, 2004. ISBN 80-7262-257-9.
- [23] E. Čech, Z. Hájek, K. Maršál, and B. Srp. *Porodnictví - 2., přepracované a doplněné vydání*. Grada, 2006. ISBN 9788024713038.
- [24] Creative Commons. *A typical CTG output for a woman not in labour*. 2010. Visited on 2015-05-10.
http://commons.wikimedia.org/wiki/File:CTG_Output.jpg.

- [25] V. Chudáček, J. Spilka, L. Lhotská, P. Janků, M. Koucký, M. Huptych, and M. Burša. Assessment of features for automatic CTG analysis based on expert annotation. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*. 2011, 6051–6054. DOI 10.1109/IEMBS.2011.6091495.
- [26] Lucie Nováková. *Ženská rozmnožovací soustava, těhotenství a porod*. 2015. Visited on 2015-04-13.
<http://fb1t.cz/?p=332>.
- [27] R. Chmel. *Otázky a odpovědi o porodu - 2., aktualizované a doplněné vydání*. Grada, 2008. ISBN 9788024721422.
- [28] T. Binder. *Peripartální krvácení*. 2013. Visited on 2015-04-02.
<http://zdravi.e15.cz/clanek/postgradualni-medicina/peripartalni-krvaceni-470634>.
- [29] ÚZIS ČR. Singleton and multifetal pregnancies according to National register of parturient mothers and National register of newborns in 2005. *Aktuální informace*. 2006, (53/06), 2.
- [30] C. Borton, and C. Tidy. *Gravidity and Parity Definitions (Implications in Risk Assessment)* — *Patient.co.uk*. 2014. Visited on 2003-03-20.
<http://www.patient.co.uk/doctor/gravidity-and-parity-definitions-and-their-implications-in-risk-assessment>.
- [31] Denise M. Main, Elliott K. Main, and Dan H. Moore. The relationship between maternal age and uterine dysfunction: A continuous effect throughout reproductive life. *American Journal of Obstetrics and Gynecology*. 2000, 182 (6), 1312–1320. DOI 10.1067/mob.2000.106249.
- [32] ÚZIS ČR. Mother and newborn 2012. *Zdravotnická statistika*. 2012, 2.
- [33] B. Bodner-Adler, K. Bodner, O. Kimberger, P. Lozanov, P. Husslein, and K. Mayerhofer. Women’s position during labour: influence on maternal and neonatal outcome.. *Wiener klinische Wochenschrift*. 2003, 115 (19-20), 720–723.
- [34] Medicabáze.cz. *Detail hesla - Kefalopelvický nepoměr*. 2007. Visited on 2015-04-10.
http://www.medicabaze.cz/index.php?sec=term_detail&categId=13&cname=Gynekologie+a+porodnictv%C3%AD&letter=K&termId=2968&tname=Kefalopelvick%C3%BD+nepom%C4%9Br.
- [35] MEFANET. *Laboratorní vyšetření acidobazické rovnováhy*. 2014. Visited on 2015-04-25.
<http://www.wikiskripta.eu/index.php/Laboratorn%C3%AD.vy%C5%A1et%C5%99en%C3%AD.acidobazick%C3%A9.rovnov%C3%A1hy>.

- [36] J. Kolodner. *Case-Based Reasoning*. Elsevier, 2014. ISBN 9781483294490.
- [37] M. Richter, and R. Weber. *Case-Based Reasoning: A Textbook*. Springer Berlin Heidelberg, 2013. ISBN 9783642401671.
- [38] Klaus D. Althoff, R. Bergmann, S. Wess, M. Manago, E. Auriol, Oleg I. Larichev, A. Bolotov, Yurii I. Zhuravlev, and Serge I. Gurov. Case-based reasoning for medical decision support tasks: The Inreca approach. *Artificial Intelligence in Medicine*. 1998, 12 (1), 25–41. DOI 10.1016/S0933-3657(97)00038-9.
- [39] Xiaolong Xie, Lin Lin, and Shisheng Zhong. Handling missing values and unmatched features in a CBR system for hydro-generator design. *Computer-Aided Design*. 2013, 45 (6), 963–976. DOI 10.1016/j.cad.2013.02.004.
- [40] R. Jagannathan, and S. Petrovic. *Dealing with missing values in a clinical case-based reasoning system*. In: *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*. 2009. 120–124.
- [41] Hugo López-Fernández, Florentino Fdez-Riverola, Miguel Reboiro-Jato, Daniel Glez-Peña, and José R. Méndez. Using CBR as Design Methodology for Developing Adaptable Decision Support Systems. *Efficient Decision Support Systems - Practice and Challenges From Current to Future*. 1997,
- [42] R. Davis. *Case-Based Reasoning: Knowledge-Based Application Systems, Spring 2004 Lecture 17*. 2004.
<https://courses.csail.mit.edu/6.871/lectures06/Lect17CBR.pdf>.
- [43] K. D. Ashley, and D. Bridge. *Case-Based Reasoning Research and Development: 5th International Conference on Case-Based Reasoning*. Trondheim: Springer, 2003. ISBN 9783540404330.
- [44] Yuan Minghai, and Xu Huanmin. *The Weight Calculation in the Case-Based Reasoning Based on Sensitivity Analysis*. In: *International Conference on Electrical and Control Engineering (ICECE), 2010*. Wuhan: IEEE, 2010. 3119–3121.
- [45] Yan Peng, and Like Zhuang. *A Case-based Reasoning with Feature Weights Derived by BP Network*. In: *Intelligent Information Technology Application, Workshop on*. 2007. 26–29.
- [46] Noha A. Elprince, and Osama M. Badawy. *Feature Weight Maintenance in Active Case-Based Reasoning System*. In: *Intelligent Data Engineering and Automated Learning, 4th International Conference*. Hong Kong: Springer, 2003.
- [47] L. Zhang, F. Coenen, and P. Leng. Formalising optimal feature weight setting in case based diagnosis as linear programming problems. 2002,

-
- [48] P. Grabusts. *The choice of metrics for clustering algorithms*. In: *Environment. Technology. Resources*. Rezekne, Latvia: 2011. 70–76. ISBN 9789984440712. http://zdb.ru.lv/conferences/3/VTR8_II_70.pdf.
- [49] O. E. Gundersen. Toward measuring the similarity of complex event sequences in real-time. *Lecture Notes in Computer Science*. 2012, 7466 LNAI 107–121. DOI 10.1007/978-3-642-32986-9_10.
- [50] P. Jaccard. *Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines*. Rouge, 1901.
- [51] J. H. Carter. What is the Electronic Health Record?. *Electronic Health Records: A Guide for Clinicians and Administrators*. 2008, 3–19.
- [52] Health Level Seven International. *HL7 Primary Standards*. 2015. Visited on 2015-04-21. http://www.hl7.org/implement/standards/product_section.cfm?section=1.
- [53] M. Huptych, and L. Lhotská (supervisor). *Multi-layer Data Model*. Ph.D. Thesis, CTU FEE. 2013.
- [54] The Open Group. *What Is SOA?*. 2013. Visited on 2015-05-01. <http://www.opengroup.org/soa/source-book/soa/soa.htm>.
- [55] P. Gotthard. *Einführung in Spring Data JPA mit Hibernate*. 2012. Visited on 2015-04-28. <http://www.patrick-gotthard.de/einfuehrung-in-spring-data-jpa-mit-hibernate>.
- [56] Pivotal Software. *Understanding REST*. 2015. Visited on 2015-04-28. <https://spring.io/understanding/REST>.
- [57] *PostgreSQL: About*. 2015. Visited on 2015-05-09. <http://www.postgresql.org/about/>.
- [58] *Spring IO Platform*. 2015. Visited on 2015-05-09. <http://spring.io/platform>.
- [59] *Hibernate ORM*. 2015. Visited on 2015-05-09. <http://hibernate.org/orm/>.
- [60] *Dozer*. 2015. Visited on 2015-05-09. <http://dozer.sourceforge.net/>.
- [61] *About Drupal*. 2015. Visited on 2015-05-09. <https://www.drupal.org/about>.
- [62] *Dygraphs*. 2015. Visited on 2015-05-09. <http://dygraphs.com/>.
- [63] *Gephi - The Open Graph Viz Platform*. 2015. Visited on 2015-05-07. <http://gephi.github.io/>.

- [64] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*. 2008, 2008 (10), P10008.
- [65] M. Bastian. *OpenOrd Layout*. 2013. Visited on 2015-05-07.
<https://marketplace.gephi.org/plugin/openord-layout/>.
- [66] Shawn Martin, W. Michael Brown, Richard Klavans, and Kevin W. Boyack. *OpenOrd: an open-source toolbox for large graph layout*. 2011.
<http://dx.doi.org/10.1117/12.871402>.

Appendix A

UML diagrams

A.1 Data model

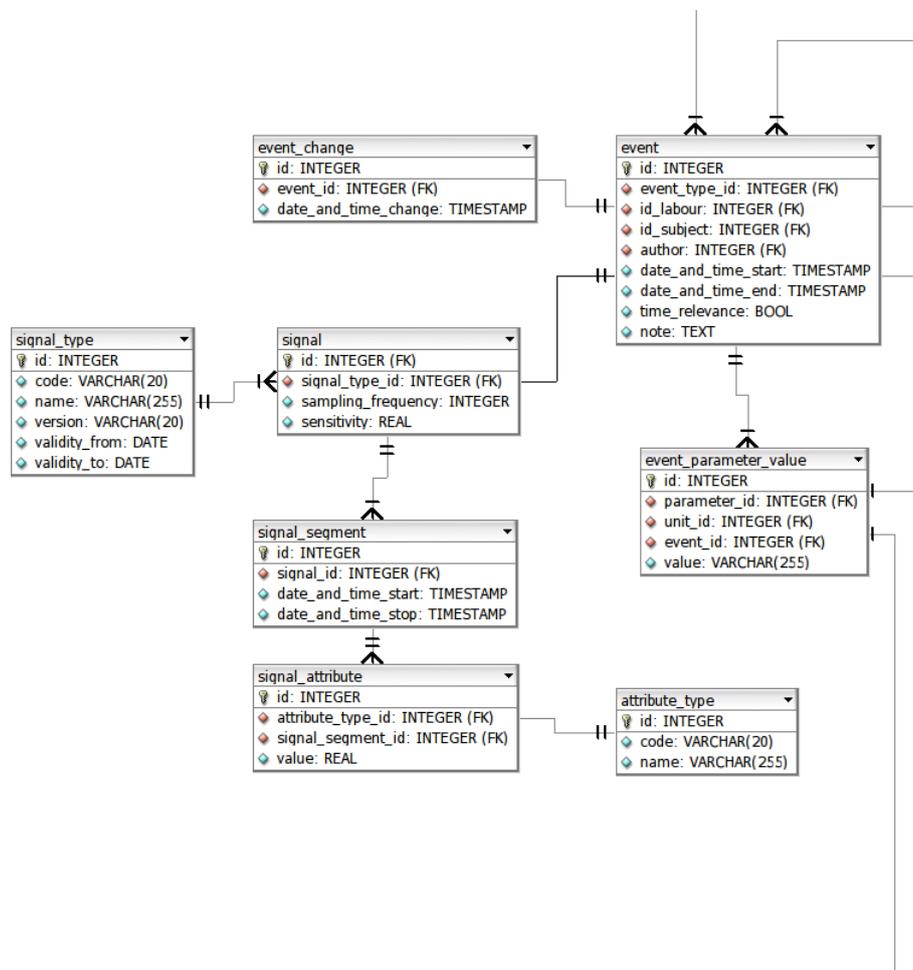


Figure A.1. ER diagram of Event data model, part 1

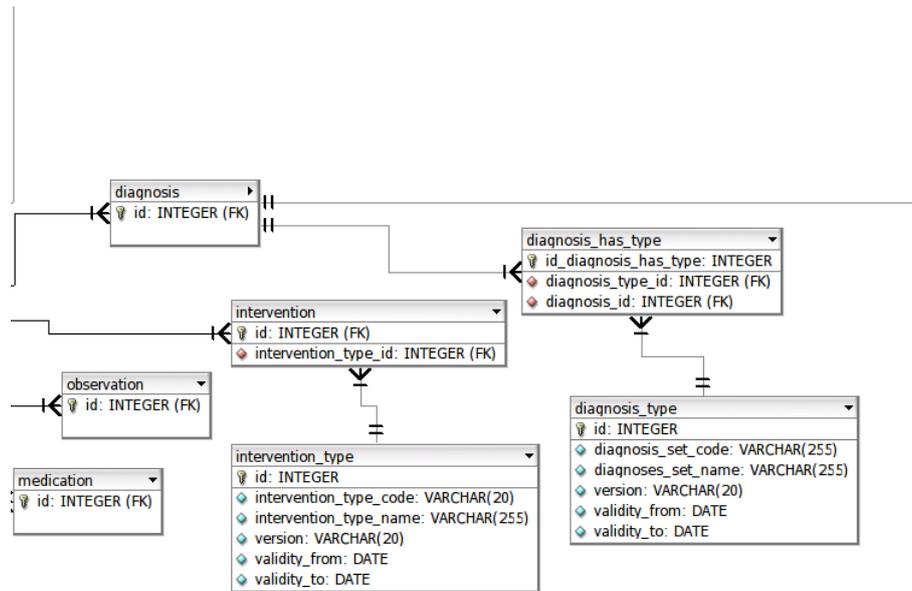
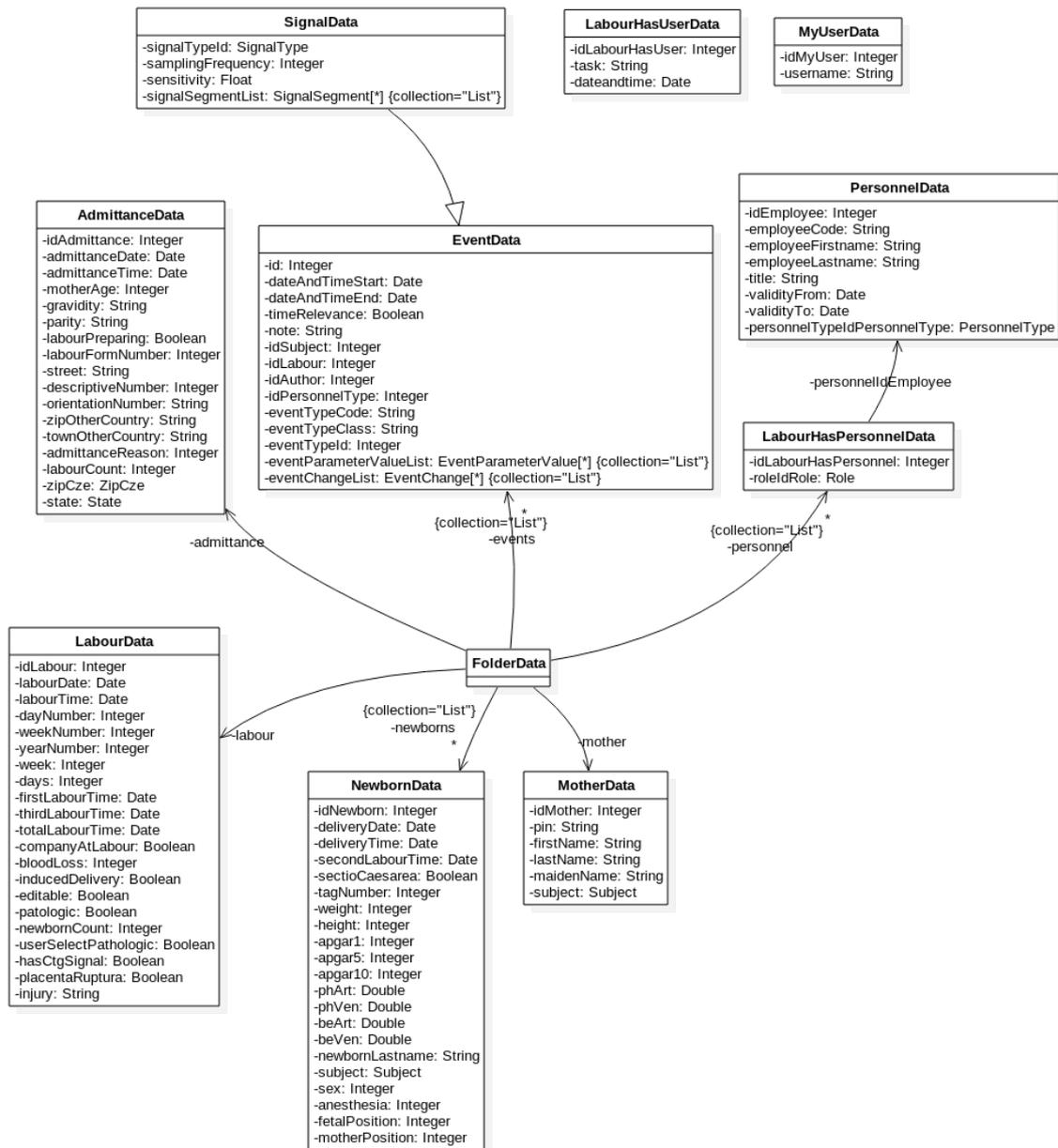


Figure A.3. ER diagram of Event data model, part 3

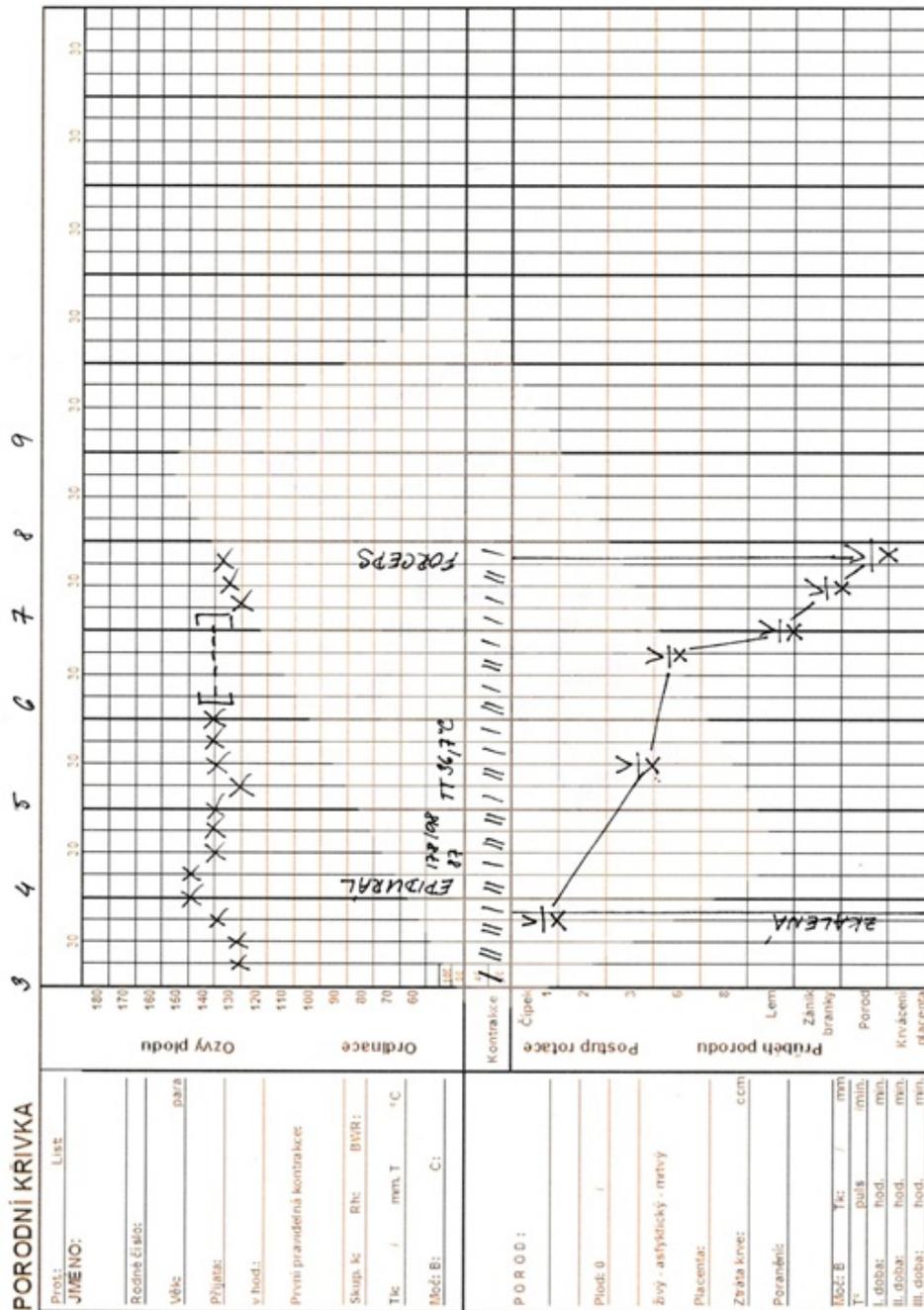
A.2 Class diagram of DTOs



Appendix B

Images

B.1 Sample partogram scan



Appendix C

Coding of attributes

attribute	code	value
intervention	1	acute caesarean section
intervention	2	primary caesarean section
intervention	2	caesarean section in anamnesis ending in c.s.
intervention	3	instrumental revision of uterus
intervention	4	fetus semi-extraction
intervention	5	fetus extraction
intervention	6	forceps
intervention	7	VEX
intervention	8	manual lysis
intervention	9	hysterectomy
intervention	10	episiotomy
intervention	11	all types of ruptures
intervention	12	labor sine

attribute	code	value
mother position	1	sitting
mother position	2	standard
mother position	3	birth on the side
mother position	4	stool
mother position	5	bag
mother position	6	standing
mother position	7	kneeling
mother position	8	squatting
mother position	9	water birth

attribute	code	value
fetal position	1	standard (head)
fetal position	2	breech birth
fetal position	3	breech birth (sectio caesarea)
fetal position	4	face
fetal position	5	transverse occiput
fetal position	6	forehead
fetal position	7	abnormal rotation
fetal position	8	top of the head
fetal position	9	asynclitism
fetal position	10	straight occiput
fetal position	11	caesarean section

attribute	code	value
sex	1	not specified
sex	2	male
sex	3	female
sex	4	can not specify

attribute	code	value
anesthesia	1	general anesthesia
anesthesia	2	epidural
anesthesia	3	spinal
anesthesia	4	none

Appendix D

Experiment samples

D.1 Experiment 1

4. Cluster (blue)

■ Case 4796 with solution 5

Labor: bloodLoss=350, pregDays=273, firstStage=300, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=34, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=40

■ Case 5005 with solution 2

Labor: bloodLoss=250, pregDays=276, firstStage=215, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=32, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10

■ Case 4858 with solution 4

Labor: bloodLoss=300, pregDays=278, firstStage=240, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=36, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=45

5. Cluster (violet)

■ Case 4769 with solution 1

Labor: bloodLoss=300, pregDays=281, firstStage=280, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=32, reason=17,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5

■ Case 4863 with solution 1

Labor: bloodLoss=350, pregDays=284, firstStage=275, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=30, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5

■ Case 5306 with solution 4

Labor: bloodLoss=300, pregDays=281, firstStage=295, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=null, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5

6. Cluster (pink)

■ Case 5057 with solution 12

Labor: bloodLoss=400, pregDays=278, firstStage=0, induced=0,
Admittance: grav=IV, par=III, abortions=1, prep=0, motherAge=35, reason=3,
Newborn: motherPos=2, fetalPos=1, anesthesia=2, secondStage=0

■ Case 5299 with solution 9

Labor: bloodLoss=500, pregDays=276, firstStage=0 induced=0,
 Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=null, reason=3,
 Newborn: motherPos=2, fetalPos=1, anesthesia=2, secondStage=0

- Case 5631 with solution 9

Labor: bloodLoss=400, pregDays=275, firstStage=0, induced=0,
 Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=null, reason=3,
 Newborn: motherPos=2, fetalPos=1, anesthesia=2, secondStage=0

D.2 Experiment 2a

0. Cluster (magenta)

- Case 5682 with solution 11

Labor: bloodLoss=300, pregDays=284, firstStage=300, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=1,
 Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=30, sex=2, weight=3790

- Case 4959 with solution 10

Labor: bloodLoss=300, pregDays=279, firstStage=285, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=28, reason=1,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3000

- Case 5343 with solution 12

Labor: bloodLoss=300, pregDays=269, firstStage=235, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=19,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3170

20. Cluster (black)

- Case 5461 with solution 10

Labor: bloodLoss=300, pregDays=280, firstStage=295, induced=0,
 Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=null, reason=17,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3630

- Case 5230 with solution 11

Labor: bloodLoss=300, pregDays=281, firstStage=95, induced=0,
 Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=26, reason=19,
 Newborn: motherPos=2, fetalPos=1, anesthesia=null, secondStage=10, sex=2, weight=3570

- Case 5389 with solution 10

Labor: bloodLoss=300, pregDays=274, firstStage=285, induced=0,
 Admittance: grav=III, par=II, abortions=1, prep=0, motherAge=null, reason=17,
 Newborn: motherPos=2, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3850

19. Cluster (cyan)

- Case 4802 with solution 12

Labor: bloodLoss=300, pregDays=281, firstStage=220, induced=0,
 Admittance: grav=V, par=III, abortions=2, prep=0, motherAge=29, reason=1,
 Newborn: motherPos=4, fetalPos=1, anesthesia=4, secondStage=10, sex=2, weight=4030

- Case 5500 with solution 12

Labor: bloodLoss=300, pregDays=285, firstStage=225, induced=0,
 Admittance: grav=IV, par=III, abortions=1, prep=0, motherAge=null, reason=1,
 Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3200

- Case 5050 with solution 11

Labor: bloodLoss=300, pregDays=282, firstStage=225, induced=0,
 Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=39, reason=1,
 Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3810

24. Cluster (pink)

■ Case 4714 with solution 2

Labor: bloodLoss=300, pregDays=261, firstStage=0, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=31, reason=3,
 Newborn: motherPos=2, fetalPos=12, anesthesia=2, secondStage=0, sex=3, weight=2050

■ Case 4806 with solution 2

Labor: bloodLoss=500, pregDays=275, firstStage=0, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=30, reason=3,
 Newborn: motherPos=2, fetalPos=12, anesthesia=2, secondStage=0, sex=3, weight=3750

■ Case 5322 with solution 2

bloodLoss=500, pregDays=272, firstStage=0, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=3,
 Newborn: motherPos=2, fetalPos=12, anesthesia=2, secondStage=0, sex=3, weight=2630

21. Cluster (blue)

■ Case 4736 with solution 1

Labor: bloodLoss=500, pregDays=283, firstStage=420, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=29, reason=1,
 Newborn: motherPos=2, fetalPos=10, anesthesia=3, secondStage=0, sex=2, weight=3560

■ Case 5062 with solution 1

Labor: bloodLoss=500, pregDays=280, firstStage=380, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=24, reason=19,
 Newborn: motherPos=2, fetalPos=10, anesthesia=2, secondStage=null, sex=2, weight=4050

■ Case 5333 with solution 1

Labor: bloodLoss=400, pregDays=285, firstStage=405, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=null, reason=1,
 Newborn: motherPos=2, fetalPos=10, anesthesia=3, secondStage=30, sex=2, weight=3600

D.3 Experiment 2b

0. Cluster (green)

■ Case 4988 with solution 10

Labor: bloodLoss=300, pregDays=283, firstStage=350, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=29, reason=1,
 Newborn: motherPos=2, fetalPos=1, anesthesia=4, secondStage=30, sex=3, weight=3580

■ Case 5000 with solution 12

Labor: bloodLoss=350, pregDays=284, firstStage=310, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=32, reason=1,
 Newborn: motherPos=2, fetalPos=1, anesthesia=2, secondStage=10, sex=2, weight=3660

■ Case 5067 with solution 10

Labor: bloodLoss=300, pregDays=280, firstStage=290, induced=0,

Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=29, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=3300

2. Cluster (red)

■ Case 5012 with solution 10

Labor: bloodLoss=300, pregDays=281, firstStage=190, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=33, reason=19,
Newborn: motherPos=2, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3500

■ Case 4862 with solution 10

Labor: bloodLoss=300, pregDays=274, firstStage=120, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=31, reason=1,
Newborn: motherPos=2, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3500

■ Case 4860 with solution 11

Labor: bloodLoss=300, pregDays=280, firstStage=340, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=31, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3250

3. Cluster (cyan)

■ Case 4875 with solution 10

Labor: bloodLoss=300, pregDays=263, firstStage=190, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=30, reason=19,
Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=15, sex=2, weight=3190

■ Case 4735 with solution 10

Labor: bloodLoss=300, pregDays=263, firstStage=140, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=25, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=2900

■ Case 5235 with solution 10

Labor: bloodLoss=300, pregDays=265, firstStage=120, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=35, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=20, sex=2, weight=2820

D.4 Experiment 3a

0. Cluster (orange)

■ Case 5599 with solution 2

Labor: bloodLoss=400, pregDays=278, firstStage=0, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=null, reason=3,
Newborn: motherPos=2, fetalPos=3, anesthesia=2, secondStage=0, sex=3, weight=3380,
Partogram: medications=[], diagnoses=[O32,O82]

■ Case 5692 with solution 2

Labor: bloodLoss=400, pregDays=271, firstStage=0, induced=0,
Admittance: grav=II, par=I, abortions=1, prep=0, motherAge=null, reason=3,
Newborn: motherPos=2, fetalPos=3, anesthesia=1, secondStage=0, sex=3, weight=3450,
Partogram: medications=[], diagnoses=[O32,O33,O82]

■ Case 5373 with solution 2

Labor: bloodLoss=300, pregDays=278, firstStage=0, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=1, motherAge=null, reason=3,

Newborn: motherPos=2, fetalPos=2, anesthesia=2, secondStage=0, sex=2, weight=3500,
Partogram: medications=[], diagnoses=[O32,O82,O99]

3. Cluster (black)

■ Case 5636 with solution 10

Labor: bloodLoss=300, pregDays=273, firstStage=165, induced=1,
Admittance: grav=II, par=I, abortions=1, prep=0, motherAge=null, reason=2,
Newborn: motherPos=2, fetalPos=1, anesthesia=4, secondStage=20, sex=3, weight=3010,
Partogram: medications=[], diagnoses=[O13,O24,O80]

■ Case 4977 with solution 7

Labor: bloodLoss=350, pregDays=283, firstStage=195, induced=1,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=27, reason=2,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=55, sex=2, weight=3000,
Partogram: medications=[], diagnoses=[]

■ Case 4855 with solution 10

Labor: bloodLoss=300, pregDays=273, firstStage=120, induced=1,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=34, reason=2,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=20, sex=3, weight=3210,
Partogram: medications=[], diagnoses=[O24,O80]

8. Cluster (magenta)

■ Case 5241 with solution 10

Labor: bloodLoss=300, pregDays=285, firstStage=235, induced=1,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=30, reason=2,
Newborn: motherPos=2, fetalPos=1, anesthesia=null, secondStage=15, sex=2, weight=4060,
Partogram: medications=[epiduralAnalgesia], diagnoses=[O48,O68,O80,O98]

■ Case 5301 with solution 10

Labor: bloodLoss=350, pregDays=267, firstStage=300, induced=1,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=15, sex=2, weight=3110,
Partogram: medications=[epiduralAnalgesia,oxytocine], diagnoses=[O42,O80]

■ Case 5455 with solution 11

Labor: bloodLoss=300, pregDays=280, firstStage=350, induced=1,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=2,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=2950,
Partogram: medications=[epiduralAnalgesia], diagnoses=[O13,O70]

D.5 Experiment 3b

1. Cluster (cyan)

■ Case 5681 with solution 10

Labor: bloodLoss=300, pregDays=264, firstStage=370, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=3220,
Partogram: medications=[epiduralAnalgesia,oxytocine], diagnoses=[O42,O80]

■ Case 5366 with solution 10

Labor: bloodLoss=300, pregDays=275, firstStage=220, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=19,

Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=3250,
Partogram: medications=[epiduralAnalgesia], diagnoses=[O80]

■ Case 4745 with solution 10

Labor: bloodLoss=300, pregDays=283, firstStage=345, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=29, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=15, sex=3, weight=4100,
Partogram: medications=[epiduralAnalgesia], diagnoses=[O80]

2. Cluster (green)

■ Case 5115 with solution 11

Labor: bloodLoss=300, pregDays=282, firstStage=205, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=32, reason=1,
Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3590,
Partogram: medications=[], diagnoses=[E03,H52,J45,O22,O70,O80]

■ Case 4760 with solution 12

Labor: bloodLoss=300, pregDays=275, firstStage=170, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=35, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3310,
Partogram: medications=[], diagnoses=[O24,O80]

■ Case 5730 with solution 11

Labor: bloodLoss=400, pregDays=279, firstStage=300, induced=0,
Admittance: grav=II, par=II, abortions=0, prep=0, motherAge=null, reason=1,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3820,
Partogram: medications=[], diagnoses=[O80]

3. Cluster (yellow)

■ Case 5285 with solution 12

Labor: bloodLoss=300, pregDays=264, firstStage=175, induced=0,
Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=null, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3290,
Partogram: medications=[], diagnoses=[O42,O69,O80]

■ Case 5405 with solution 12

Labor: bloodLoss=300, pregDays=271, firstStage=240, induced=0,
Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=null, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=3060,
Partogram: medications=[], diagnoses=[O42,O80]

■ Case 4990 with solution 11

Labor: bloodLoss=300, pregDays=267, firstStage=155, induced=0,
Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=29, reason=19,
Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=5, sex=2, weight=2720,
Partogram: medications=[], diagnoses=[I51,O42,O69,O70,O80]

D.6 Experiment 4

16. Cluster (black)

■ Case 4952 with solution 10

Labor: bloodLoss=300, pregDays=269, firstStage=490, induced=0,
Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=21, reason=1,

Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=20, sex=2, weight=2750,
 Partogram: medications=[], diagnoses=[F17,O80], cervixDilation=null, contractions=null,
 CTG: baseline=null, accelerations=null, decelerations=null

- Case 4828 with solution 12

Labor: bloodLoss=300, pregDays=283, firstStage=275, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=28, reason=19,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=3, weight=3800,
 Partogram: medications=[epiduralAnalgesia], diagnoses=[O80], cervixDilation=null, contrac-
 tions=null,
 CTG: baseline=null, accelerations=null, decelerations=null

- Case 4933 with solution 11

Labor: bloodLoss=300, pregDays=284, firstStage=315, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=23, reason=1,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=20, sex=3, weight=3360,
 Partogram: medications=[epiduralAnalgesia], diagnoses=[O71,O80], cervixDilation=null, con-
 tractions=null,
 CTG: baseline=null, accelerations=null, decelerations=null

2. Cluster (violet)

- Case 5768 with solution 12

Labor: bloodLoss=300, pregDays=274, firstStage=220, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=null, reason=1,
 Newborn: motherPos=3, fetalPos=1, anesthesia=4, secondStage=5, sex=3, weight=3570,
 Partogram: medications=[], diagnoses=[O80], cervixDilation=null, contractions=null,
 CTG: baseline=null, accelerations=null, decelerations=null

- Case 4746 with solution 11

Labor: bloodLoss=300, pregDays=281, firstStage=270, induced=0,
 Admittance: grav=I, par=I, abortions=0, prep=0, motherAge=28, reason=19,
 Newborn: motherPos=1, fetalPos=1, anesthesia=4, secondStage=10, sex=2, weight=3330,
 Partogram: medications=[], diagnoses=[O70,O71], cervixDilation=null, contractions=null,
 CTG: baseline=null, accelerations=null, decelerations=null

- Case 5533 with solution 10

Labor: bloodLoss=200, pregDays=279, firstStage=190, induced=0,
 Admittance: grav=III, par=III, abortions=0, prep=0, motherAge=null, reason=1,
 Newborn: motherPos=1, fetalPos=8, anesthesia=4, secondStage=5, sex=3, weight=3420,
 Partogram: medications=[], diagnoses=[O69,O80], cervixDilation=null, contractions=null,
 CTG: baseline=null, accelerations=null, decelerations=null