Methods, architecture, evaluation and usability of a case-based antibiotics advisor

Lothar Gierl *, Dagmar Steffen, Dusan Ihracky, Rainer Schmidt

Institute for Medical Informatics and Biometry, University of Rostock, Rembrandtstrasse 16/17, D-18055 Rostock, Germany

Received 12 February 2002; received in revised form 15 July 2002; accepted 29 July 2002

Abstract

In this paper, we discuss the usability of an antibiotics therapy adviser, with a broad, complex spectrum of functions which we have developed within the ICONS project. We present the architecture of the system, case-based reasoning methods used, steps and results of medical evaluations, which are concerning the quality of the recommended therapies, the user friendliness of the system and the interpretation of laboratory results. Furthermore, we discuss problems of transferability of such a system from one site to another as well as problems of local susceptibility patterns and individual dose regimens.

* Corresponding author

E-mail addresses: lothar.gierl@medizin.uni-rostock.de (L. Gierl), rainer.schmidt@medizin.uni-rostock.de (R. Schmidt).

Keywords: Antibiotics; Adviser; Case-based reasoning; Evaluation; Resistance surveillance

1. Introduction

Severe bacterial infections are still a life threatening complication in intensive care medicine correlated with a high mortality. Identification of bacterial pathogens is often difficult. It normally requires at least 24 h to identify the pathogen that is responsible for an infection and at least another 24 h to find out which antibiotics have therapeutic effects against the identified pathogen. To not endanger the patient, physicians often have to start an antimicrobial therapy before the responsible pathogen and its sensitivities are determined [1].

This sort of antibiotic therapy is called “calculated” in contrast to a “selective” therapy, which is used when microbiological results are already available. Empirical results [2] indicate that early antimicrobial therapy affects survival of patients.

A study [3] showed that computer-assisted decision support programs can improve antibiotic use, reduce associated costs and stabilize the emergence of antibiotic-resistant pathogens. The study revealed antibiotic costs per treated patient of $51.90 in 1994 in USA. Moreover, it was found that the use of antibiotics could be decreased by 22.8%, while the proportion of hospitalized patients who received antibiotics increased during the study. Experiences show routine prescription of antibiotics given before many operations prophylactily (frequently even Vancomycin) is not indi-
icated. The duration of the regimen is often too long, namely up to 10 days, which is without any use.

Experiences of other sites and the necessity of controlling resistance development [4] encouraged us to extend a computer program, called ICONS, for antibiotic therapy advice, resistance control and individual dose prescription.

In the framework of the MEDWIS project of the MEDIS institute of GSF, partly funded by the German Ministry for Research and Technology, between 1993 and 1996 we developed a first version of ICONS. The main task of ICONS is to present suitable calculated antibiotics therapy advice for intensive care patients, who have developed a bacterial infection as an additional complication. Since for such critical patients physicians cannot wait for the laboratory results, we use an expected pathogen spectrum based on medical background knowledge. This spectrum should be completely covered by each advisable antibiotics therapy. Furthermore, as advice is needed very quickly, we speed-up the process of computing advisable antibiotics therapies by using case-based reasoning (CBR) methods. We search for a similar previous patient and transfer the suggested therapies made for his situation to the current patient. These previous suggestions are adapted to be applicable to the new medical situation of the current patient.

A first evaluation showed that such a system could be helpful at least in ICUs in selecting medically and economically appropriate antibiotics regimen. Background of this evaluation was that the spectra of pathogens and their resistances could differ widely from hospital to hospital. Already de Dombal [5] showed that sets of cases from different medical sites in Europe could have a significant impact on the performance of a knowledge-based system if the knowledge base were not adapted to the local case-mix. Local case-based knowledge should, therefore, be considered.

The knowledge on organ-specific pathogen spectra gained from literature was evaluated by the retrospective analysis of the microbiological data of the anaesthesiological and the neurosurgical ICU of the Medical School of the University Munich of the years 1994 and 1995. 14,800 microbiological culture results of consecutive 750 patients were analyzed statistically for frequency and antibiotic sensitivity of pathogens in relation to the concerned organ (systems). Seventy-five percent of the culture results did not show any (relevant) pathogen growth. In 25%, one or more pathogens could be cultured. These pathogens were assigned to the relevant organs depending on the materials they were cultured from. The local pathogen spectra for the different organs gained this way were compared with those found in literature. For some organs—especially of immunocompromized patients—the literature-based pathogen spectra were modified. In addition, the sensitivity for antibiotics of relevant bacterial pathogens were checked and also compared with the literature findings. Partly, significant differences could be observed, which again supports the necessity of local adaptation. After having modified our knowledge base to the local conditions, a long prospective evaluation of ICONS for medical correctness and technical practicability under clinical conditions has been started.

To assess the correctness of the therapy advice, the user friendliness and the user–machine interaction, we designed a questionnaire. Five experienced physicians of the anaesthesiology ICU filled in the questionnaires. The questionnaire encompasses typical cases for which a calculated antibiosis should be determined. It contained 14 questions. The physicians were trained in using ICONS. They applied the same cases in ICONS. The advice of the system had to be assessed. There were further questions on user friendliness, clinical relevance and usability to answer. The results in regard to usability and method of ICONS encouraged us to extend the functionality of the system and to further evaluate it at another medical site. We wanted to investigate what architecture for an antibiotics advisor was necessary to be used routinely in a medical environment.

All these investigations and our experiences on knowledge-based systems in other medical domains lead us to use CBR as a technology of an antibiotic advisor. We found that especially for diagnostic tasks medical CBR systems should take the case-oriented reasoning of physicians into
account [6]. Such systems should not only consist of general medical domain knowledge plus a flat case base, but the case base should be structured by prototypes [7]. Prototypes are a generalization from single to clustered typical cases. The main purpose of such generalized knowledge is to guide the retrieval process and to decrease the amount of storage by erasing redundant cases. In domains with rather weak domain theories, another advantage of case-oriented techniques is their ability to learn from cases. Only gathering new cases may improve the systems ability to find suitable similar cases for current problems, but it does not elicit the intrinsic knowledge of the stored cases. To learn the knowledge contained in cases, a generalization step is necessary. Since physicians reason with prototypical and exceptional cases anyway, the creation of prototypes seems to be an adequate learning technique at least for medical domains.

2. Background

2.1. Own prior work on CBR systems in medicine

In the recent years, we have build four medical CBR systems. On various occasions, we have presented them [8–10] and have discussed special aspects like the role of prototypes [11] and the similarity problem for medical applications [12].

A first version of an early warning system concerning the kidney function [10] was implemented. The system COSYL [8] aims at the support of an outpatient clinic for liver-transplanted patients.

In the last years, we investigated the usability of ICONS along the parameters user friendliness, quality of recommended therapies and interpretation of laboratory results.

Furthermore, we were confronted with problems of transferability from one site to another.

2.2. Related work on systems supporting antibiotic therapy or pathogen surveillance

Recently, some other systems have been developed which support antibiotic therapy. The medical information system of Slovenia provides a graphic presentation of the current local resistance development of nosocomial pathogens [14]. The database management system AntibiosIS [15] offers information about interactions, advisable antibiotic combinations, reactions, side effects, indications and contraindications.

However, apart from ICONS only very few systems like ANTICIPATOR [16] and Evans’s program inside the HELP system [17] give comprehensive decision support. Like ICONS, both systems lay special emphasis on the problem of covering the calculated pathogen spectrum with antibiotic combinations. ICONS and ANTICIPATOR pay much attention to changes of resistances. This is a lack of Evans’s system. ANTICIPATOR does not consider earlier cases with infectious diseases. Evans’s system analyses previous cases and determines each month the five most frequent pathogens, which constitute the current pathogen spectrum. ICONS determine calculated pathogen spectra specific to the organ system where the infection occurred. ANTICIPATOR and Evans’s system deal with contraindications and side effects only cursorily. In ICONS, these restrictions are considered adequate to their different importances for the therapy advice.

3. Design considerations

3.1. Case-base reasoning

The core of the system is a CBR method. CBR has become a successful technique for knowledge-based systems in many domains. CBR means to use previous experience in form of cases to understand and solve new problems. A case-based reasoner remembers former cases similar to the current problem and attempts to modify their solutions to fit for the current case. Fig. 1 shows the CBR cycle developed by Aamodt and Plaza [18]. The underlying idea is the assumption that
similar problems have similar solutions. Though this assumption is not always true, it holds for many practical purposes.

CBR consists of two main tasks. The first is the retrieval, which is the search for or the calculation of most similar cases. If the case base is rather small, a sequential calculation is possible, otherwise faster non-sequential indexing [19] or classification algorithms (e.g. the ID3 family [20] or nearest-neighbour match [21]) should be applied. For this task, much research has been undertaken in the recent years and actually it has become correspondingly easy to find sophisticated CBR retrieval algorithms adequate for nearly every sort of application problem. The second task, the adaptation (reuse and revision) means a modification of solutions of former similar cases to fit for a current one. If there are no important differences between a current and a similar case, a simple solution transfer is sufficient. Sometimes only few substitutions are required, but sometimes the adaptation is a very complicated process. So far, no general adaptation methods or algorithms have been developed. The adaptation is still absolutely domain-dependent.

Similarity-based learning, introduced by Winston [22], developed to handle with many cases in memory, was a precursor of CBR.

A case is a record of an episode where a medical problem is solved. Following Richter [23], a case in its abstract form is represented as an ordered pair (problem, solution). More specifically, we will state that a case base (CB) is defined as follows. CB:: = \{C_1, \ldots, C_n\}; C_i are cases, where C_i:: = \{s_1, \ldots, s_{m_i}; t_j\} is a case; s_i are features and symptoms of a case (the problem); t_j is the class j of a solution connected with C_i; T:: = \{t_1, \ldots, t_c\} is the set of classes of solutions of our medical domain (antibiotics therapies).

CBR is a methodology for reasoning and learning. A case-based reasoner learns in two ways. First, it can become a more efficient reasoner by remembering old solutions and adapting them. Second, it becomes more and more competent, because of its increasing experience [18].

The difficulties of this method are to determine the similarity of cases, to efficiently guide the search for similar cases, and to adapt the old solution to the new problem. Solving the similarity problems means to determine the most similar case in memory to the new problem. All the cases in memory have to be compared with the new one by using a measure of similarity. This may be Tversky’s [24] contrast model of features or Rosch’s [25] model of category resemblance or other similarity measures. Sometimes the features are weighted. The weight for one feature depends on the importance of the feature to the problem it belongs to. Typically in domains with weak theory no possibility exists to derive these weights. Categories like importance, necessity, typicality of the problems features might be used instead of empirically measured weights.

In general, we had to state that similarity measures are not metrics. They only are quasi-metrics, because the separation axiom for metric spaces does not hold (Hausdorff spaces). This means we cannot assume that two points in our space of cases CB are closed disjoint subsets. This could be seen easily if we look, for instance, at two different cases of infections where both could be caused by the same pathogen. So, the following axioms hold for similarity measures where d is a similarity function: There is always \(d(a,a) = 1\). For two cases \(a,b\) holds \(d(a,b) = d(b,a)\). For the values of \(d\) holds \(0 \leq d(a,b) \leq 1\). For three cases \(a, b, c\) holds \(d(a,c) \leq d(a,b) + d(b,c)\). If \(a \neq b\), then \(d(a,b) > 0\).
To increase the performance of a case-based system, it is necessary to use a memory structure that avoids determining the similarities for all cases in memory. Therefore, it is useful to classify the cases in a hierarchical categorization tree. The leaves are the cases, at the next higher level are the prototypes of solutions and the higher levels of the hierarchy are categories that may be ordered hierarchically. In the first step, the similarities for only the direct successor of the root are calculated. Only the subtree with the greatest similarity must be inspected further. This leads the search through the tree down to a prototype. This prototype possesses all the typical features of his cases. The prototype is a generalization of his cases.

3.2. Why CBR for medical decision making?

Especially in medicine, the knowledge of experts does not only consist of rules, but of a mixture of knowledge from textbooks, journals and experience. The latter consists of cases, typical and exceptional ones, and the reasoning of physicians takes them into account [6]. In medical knowledge-based systems, there are two sorts of knowledge: objective knowledge, which can be found in textbooks, and subjective knowledge as experience which is limited in space and time and changes frequently.

The problem of updating the changeable subjective knowledge can partly be solved by incrementally incorporating new up-to-date cases [12]. Both sorts of knowledge can clearly be separated. Objective textbook knowledge can be represented in form of rules or functions, while subjective knowledge is contained in cases.

Knowledge acquisition is the bottleneck of building expert systems. Especially in weak or unstructured domains, it is difficult to express the experts knowledge into a set of rules. The method of CBR might be a contribution to partly solve this problem.

So, the arguments for case-oriented methods are as follows:

1) Reasoning with cases corresponds with the decision-making process of physicians.
2) Incorporating new cases means automatically updating parts of the changeable knowledge.
3) Revision including forgetting knowledge is part of the architecture.
4) Objective and subjective knowledge can be clearly separated.
5) Adapting CBR systems to a special medical facility is easy.
6) As cases are routinely stored, integration into clinic communication systems is easy.

3.3. Resistance surveillance as part of an antibiotics strategy

Using antibiotics with low probability to meet a resistant strain of the pathogen of the patient is crucial especially in a calculated therapy regimen. Prescribing antibiotics without ward-specific knowledge on the resistant situation is of minor use. An antibiotics advisor, therefore, has to adapt its internal knowledge automatically depending on the resistance situations of the ward where a patient lies and the clinic as a whole.

Using the resistance table or graphic screen from commercial microbiologic information systems and sharing this information is a very convenient method to overview the specific resistance situation of a ward. But tracking resistance in space and time over the whole clinic is difficult but necessary, however [4]. Resistant agents could float between wards, ICUs or operating theatres. This leads to the requirement to present this pattern in a 3D way moving in time. The microbiologists could decide what antibiotics should be used at all to attack a special pathogen with regard to its resistance abilities. A prerequisite of resistance surveillance is a steady flow of antibiogram data of a computerized microbiology system. Therefore, we had to solve the communication problem of such data concerning data confidence, reliability and consistency.

3.4. Individualized dosage

Usually antibiotics dosage will be prescribed concerning age and weight of patients. Only for a small population of patients (immunocompromized) an individual dosage will be calculated
depending on measured levels of this antibiotic in blood. Therefore, ICONS has a component to individually prescribe antibiotic dosage using a one-compartment pharmacokinetic model.

4. System description

4.1. Overview

An overview of the system is presented in Fig. 2. ICONS consists of a kernel module to support antibiotics therapy. A resistance surveillance module (RESISTANCE) provides the pattern of susceptibility of pathogens over time in linear graphics (used at wards) as well as a presentation of this data in space and time (will be used at the microbiology department). Antibiograms and data of patient stay were sent via a communication server to the ICONS server. The module for individualized dosage (ABINI) is for the pharmacology department.

Details on evaluation and experience concerning the methods employed in ICONS could be found in [26,27].

4.2. Strategy for selecting advisable antibiotic therapies

As ICONS is not a diagnostic system, we do not attempt to deduce evidence for diagnoses based on symptoms, frequencies or probabilities, but instead pursue a strategy that can be characterized as follows (Fig. 3). Find all possible solutions and reduce them using the patient’s contraindications and the complete coverage of the calculated pathogen spectrum (establish-refine strategy).

First, we distinguish among different groups of patients (infection acquired in or outside the ward, respectively, the hospital, immunocompromized patients, etc.). A first list of antibiotics is generated by a susceptibility relation that for each group of pathogens provides all antibiotics which usually have therapeutic effects. This list contains all those antibiotics that can control at least a part of the potential pathogen spectrum. We obtain a second list of antibiotics by reducing the first one by applying two constraints (Fig. 4): the patient’s contraindications and the desired sphere of activity. Using the antibiotics of this second list, we try to find antibiotics, which under consideration of the expected susceptibility cover the whole pathogen spectrum individually.

Fig. 2. Overview of the modules in ICONS.
Except for some community-acquired infections, monotherapies have to be combined with synergistic or additive effecting antibiotics. If no adequate single therapy can be found, we use combination rules to generate combinations of antibiotics. Each possible combination must be tested for the ability to cover the whole-expected spectrum.

Using the antibiotics of this second list, we apply rules to generate appropriate antibiotics therapies. Each possible therapy is tested for the ability to cover the whole-expected spectrum. Before the user decides to use one of the presented therapies, he can investigate potential side effects of the antibiotics. Moreover, he may obtain information about the considered pathogen spectrum, about the daily costs of each proposed therapy, and about the current resistance situation and development. After the physician has chosen one therapy, ICONS computes the recommended dosage. Fig. 5 collects these features embedded in the flow of information.

![Fig. 3. Antibiotics selection strategy.](image)

<table>
<thead>
<tr>
<th>Advisable Therapies</th>
<th>Price (DM/Day)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LINCOSAMIDE</strong></td>
<td></td>
</tr>
<tr>
<td>Clindamycin</td>
<td>92 to 205</td>
</tr>
<tr>
<td>Ciprofloxacin</td>
<td></td>
</tr>
<tr>
<td><strong>PENICILLINE</strong></td>
<td></td>
</tr>
<tr>
<td>Piperacillin</td>
<td>48 to 119</td>
</tr>
<tr>
<td>+ Gentamicin</td>
<td></td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>65 to 146</td>
</tr>
<tr>
<td>+ Amikacin</td>
<td>166 to 238</td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>66 to 122</td>
</tr>
<tr>
<td>+ Amikacin</td>
<td>168 to 209</td>
</tr>
<tr>
<td>+ Gentamicin</td>
<td>11 to 15</td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>28 to 42</td>
</tr>
<tr>
<td>+ Amikacin</td>
<td>129</td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>75 to 140</td>
</tr>
<tr>
<td>+ Gentamicin</td>
<td>177 to 220</td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td></td>
</tr>
<tr>
<td>+ Amikacin</td>
<td></td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td></td>
</tr>
<tr>
<td>+ Gentamicin</td>
<td></td>
</tr>
<tr>
<td><strong>CIPROFLOXACIN</strong></td>
<td></td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>106 to 240</td>
</tr>
<tr>
<td>+ Gentamicin</td>
<td>85 to 159</td>
</tr>
<tr>
<td>+ Tobramycin</td>
<td>68 to 152</td>
</tr>
</tbody>
</table>

![Fig. 4. Presentation of advisable antibiotics combinations on screen (second list of antibiotics).](image)
4.3. Retrieval

One main effect in using CBR methods is to speed-up the process of finding adequate therapies. We shorten the above-described strategy (Fig. 5) of selecting advisable antibiotic therapies by searching for a similar case, retrieving its suggested therapies and by adapting them concerning the contraindications of the current patient. The retrieval consists of three steps (Fig. 6).

First we select that part of the case base, in which all cases share two attributes with the current patient: The group of patients and the infected organ system. This means a selection of the appropriate prototype tree (see Section 4.5). Subsequently, we apply the Hash-tree retrieval algorithm of Stottler et al. [19] for nominal-valued contraindications and the similarity measure of Tversky [24] for few integers-valued contraindications. Furthermore, we use an adaptability criterion, because not every case is adaptable [28]. As the attributes used for the retrieval are the contraindications, which work as constraints for the set of possible antibiotics suggestions, it is obvious that a former case that has contraindications which the current patient does not share should not be used. To guarantee this condition, the adaptability criterion has to be checked during the retrieval. This can be considered as an example to support the ideas of Smyth and Keane [28] that the similarity assumption alone is often inappropriate and that the retrieval should take the adaptation into account.

4.4. Adaptations

4.4.1. General adaptation strategy

In the antibiotics therapy adviser, three different ways of adaptations occur (Fig. 7). (1) A CBR adaptation to obtain sets of calculated advisable therapies for current patients (Fig. 4 shows the presentation of such a set). (2) An adaptation of chosen therapies to laboratory findings in regard to the current patients in individual infection situation. (3) An adaptation of the global resistance situation in the memory of background knowledge concerning the information on resistance of infections agents on a clinic level to be used in the decision support system to decide which antibiotic is not selectable regarding a special agent.
4.4.2. Adaptation regarding clinical signs

Each contraindication restricts the set of advisable therapies. As already mentioned above, we use a criterion during the retrieval which guarantees that the retrieved case does not have any additional contraindications in comparison with the current case. Otherwise the solution set for the current case would be inadmissibly reduced by additional contraindications of a previous case.

The adaptation of a previous similar case is rather simple. It is just a transfer of the set of advisable therapies and if necessary, a subsequent reduction of this set by additional contraindications of the current case.

Fig. 7. Adaptation in ICONS.
4.4.3. Adaptations of chosen therapies to laboratory findings

Adaptations of laboratory findings do not really belong to the CBR paradigm, but they are based on information about cases. The goal of the main part of the therapy adviser is to present advisable therapies before the results of the laboratory are known. When later on these results are known, the already started therapy has to be adapted to them. There are two sorts of findings: after 24 h the identification of the pathogen which is responsible for the infection and after another 24 h the sensitivity test results (antibiogram) of this pathogen against the various antibiotics. If the identified pathogen does not belong to the considered calculated pathogen spectrum and if this pathogen is according to the systems sensitivity information not sensitive against the already started therapy, new specific advisable therapies against this pathogen have to be computed. If the laboratory sensitivity test results show that the identified pathogen is in contrast to the systems sensitivity information not sensitive against the already started therapy, it leads to the same task: new “selective” advisable therapies which have therapeutic effects against the identified pathogen have to be computed.

It might seem to be a contradiction that laboratory tests can show that a pathogen is not sensitive against an antibiotic although the current sensitivity information says it should be. However, as pathogens are never exactly alike, but always slight mutations, the sensitivity information are based on a percentage value. For example, most of the problematic pathogens are nowadays only in slightly more than 80% of the cases sensitive against the strongest antibiotics. So an observed sensitivity higher than 66% is usually already considered as sensitive.

4.4.4. Adaptation of general resistance situation of the clinic

When new cases are incorporated into the system, the sensitivity information has to be updated, the laboratory findings for these new cases must be taken into account. Additionally, the used expected pathogen spectra might change on time too. For both laboratory information sources used by the system, we have implemented a periodical update. This can be seen as another form of adaptation, which is not founded on single cases, but on statistical evaluation of specific information of a number of cases.

4.5. Prototypes

Since in an incrementally working system the number of cases increases continuously, storing each case would slow down the retrieval time and exceed any space limitations. So, we decided to structure the case base by prototypes and to store only those cases that differ from their prototype significantly. Though the general use of prototypes was early introduced in the CBR field [29], it is still mainly applied in the medical domain (e.g. [30–33]). Our prototype architecture is mainly based on experience with a diagnostic application [13], where we create prototypes that share most features with most of their cases. This idea is based on empirical research [34], which indicates that people consider cases to be more “typical” when the number of features between the presented case and the “normal” case increases.

In diagnostic applications, prototypes correspond to typical diseases or diagnoses. So, for antibiotic therapies prototypes are expected to correspond to typical antibiotic treatments associated with typical clinical features of patients. However, as the attributes are contraindications, which are not responsible for the generation but the restriction of the solution set, this is only partly true. We have investigated the growth of a hierarchical prototype structure built up from a randomly ordered stream of cases.

4.5.1. Selection of a prototype tree

We do not have just one prototype tree, but a wood of trees, which are independent from each other. For each affected organ combined with each group of patients, a own tree can be generated. That means, for nearly 20 organ systems and five patient groups nearly 100 prototype trees are possible. We generate them only if required dynamically. For example, a tree for “community-acquired kidney infections” will be generated as soon as the first data input of a patient occur.
who has a kidney infection which he acquired outside the hospital.

So, all cases within the same prototype tree belong to the same group of patients, the same organ system is affected and therefore the same expected pathogen spectrum deduced from background knowledge has to be covered. The cases within a prototype tree are only discriminated from each other by their different contraindications. These are antibiotic allergies, reduced organ functions (e.g. kidney and liver), specific diagnoses (e.g. acoustic distortion or diseases of the central nervous system), special blood diseases, pregnancy and the patient’s age.

4.5.2. Generating prototypes

First, all cases are stored below the prototypes they belong to. If after storing a new case below a prototype the threshold “number of cases” is reached, the prototype will be “filled”. This means that every contraindication which occurs in the cases belonging to this prototype at least as often as the second threshold “minimum frequency” will be included into the prototype. Subsequently, the “filled” prototype can be treated like a case. The same as for cases holds for prototypes. Each contraindication restricts the set of advisable therapies. The contraindications of a prototype are those that occur most often within its cases. So, from the viewpoint of frequency they are the typical ones. Those cases that have no additional contraindications in comparison with their prototypes are erased. Only information about their occurred contraindications are saved in the frequency table of their prototype.

When later on a new case is added to an already filled prototype, its frequency table, which contains information about the frequency of the contraindications of its cases, has to be updated and if necessary the contraindications of the prototype have to be recomputed. If the (re-)computed contraindications of the prototype change, the suggested antibiotic therapies have to be recomputed too. All cases must be inspected again for their need to be stored.

Below an already existing prototype, we create a “alternative” prototype if for the latter enough cases exist (that means the threshold “number of cases” is reached) that have at least one contraindication in common, which the already existing prototype does not include. We construct this new alternative prototype from those cases that share at least one from the already existing prototype deviating contraindication. We place this new prototype in the hierarchy directly below the existing prototype (a part of a possible prototype hierarchy is shown in Fig. 8). New “alternative” prototypes differ from their superior prototypes by their contraindications and therefore by their set of advisable antibiotic therapies too.

Even the adaptation of an “empty case”, i.e. a case without any contraindication and therefore with an unrestricted set of advisable therapies, works faster than the normal program flow. The most time-consuming step of the program flow without CBR is the computation of advisable antibiotic combinations, because a lot of conditions have to be checked, e.g. the current sensitivity situation for each pathogen in the expected pathogen spectrum has to be considered for each antibiotic of each possible therapy combination. So, when the first prototype of a tree is filled, we additionally generate an artificial “empty case”, which can be retrieved if no adaptable case can be found in this tree.

4.6. Resistance surveillance

The assessment of ICONS’ advice for antibiotics therapy (see below) conducted in the Institute for Microbiology of our Medical School lead to a split of our system. As a consequence of the high division of labor ICONS was extended by a system called RESISTANCE for wards. It is an add-on of the half-year reports on antibiotics resistance some wards receive routinely from the Institute for Microbiology of our Medical School. The reports were automatically compiled by the computer system of the microbiological laboratory. RESISTANCE allows an online 1-month presentation of the resistance situation of a clinic selectively for each ward. Fig. 9 shows an example of the user interface for wards. RESISTANCE is a JAVA applet.

The second consequence is that tracking resistance at a clinic level needs functions to present it
in space and time. We implemented a 3D virtual reality system, RESIS-3D [35] to be used at the Institute for Microbiology.

Fig. 10 shows the coarse 3D picture of our clinic. Each ward in ICONS is a rough segment. Colors represent the degree of the resistance of a selected agent against selected antibiotics. The user is able to zoom-in and -out the 3D picture, and move it along the three axes. The flow of time is represented by presenting the resistance situation in time intervals backward or forward like a video clip.

The system has been implemented in VRML and JAVA. An overview of the architecture and the integration of RESIS-3D are shown in Fig. 11.

The data for both systems RESISTANCE and RESIS-3D were transmitted automatically from the Microbiology System via our communication server “egate”. The data will be coded and made anonymous. One important problem of consistency of this data is to delete records for patients with multiple investigations but the same pathogen between admission and discharge. The necessary data for this decision are part of the antibiogram.
data and have been routinely sent from the Hospital Information System to the Microbiology System.

4.7. Individualized dosage

The doses presented by ICONS are standard doses for children and adult patients depending on their weight and height. However, especially immunocompromized patients need a more individualized dose of antibiotics. A further consequence of the evaluation was, therefore, that we implemented a subsystem called ABINI to calculate individualized doses. It uses a one-compartment pharmacokinetic model of antibiotic therapy, because usually in clinical routine the method to compute individual doses has to rely on only two measured values of the antibiotics peak and through level.

5. Status report

5.1. Assessment of quality of the system’s advice

First, we have evaluated the expected pathogen spectra and the resistance situation the system starts with. We analyzed the microbiological results of 2 years provided by the microbiological laboratory of the University of Munich. For testing the correctness and the quality of the proposed therapies, we asked some experienced intensive care physicians to assess ICONS’s proposals using a questionnaire. As the inter-observer variability among the physicians concerning the assessments of ICONS’s therapy advice and concerning their own proposed therapies was tremendous (some physicians thought very highly of ICONS’s advice while others assessed some proposed therapies as unsuitable), it is impossible to define any “golden standard”.

The results show that ICONS performs comparatively well, but the transmission of the system to another site (in our case from Munich to Rostock) causes decreased correctness of the system. Therefore, we adapted the clinically used antibiotics, site-specific resistance and the regimen of combinations of antibiotics in ICONS to the clinical use in Rostock.

Partly different antibiotics are made available on the wards. This does not only concern different labels for nearly the same antibiotic provided by different companies, but also a few completely different antibiotics. So, we had to remove few antibiotics from the knowledge base and had to
put in a few others instead. And of course the rules, which contain information about which antibiotics should be combined with each other, also had to be modified. Furthermore, an antibiotics therapy adviser has to consider local characteristics, e.g. the resistance situation is mainly general according to the information from literature, but has to be adapted to local characteristics automatically.

We started with an examination of 20 cases of two structural equivalent intensive care units in the Medical School of the University of Rostock (internal medicine and surgery). In the first step, we have checked if the therapies introduced by the physicians were recommended by the updated ICONS too.

The coverage that the introduced therapies were also recommended by ICONS is about 70%. The confidence interval of the probability for matching the physician’s opinion was $45 \leq P \leq 88$ ($\%$). Using single therapies while ICONS recommended them in combination with antibiotics that have additive or synergistic effects the probability was about 12%. Nearly 18% of the cases the physicians prescribed therapies were not recommended by ICONS. The 12% combinations instead of single therapies are not a problem, because ICONS simply prefers to be on the safer side concerning a spectrum of pathogens. For those therapies ICONS did not recommend, it was difficult to decide if it were good ones or not, because we did not have any information about the quality of prescribed therapies. One idea might be to look at the success. However, for already critically ill patients, who additionally catch an infectious disease, there may be many possible explanations for a therapeutic failure. And vice versa, in case of success there might have been much better therapies available instead of the chosen ones. So, we tried to discuss with the physicians who prescribed these therapies about their reasoning. One problem with the retrospective cases was that the physicians sometimes had difficulties to remember all details if the patient record contained no information on that problem. Another problem was to assess the justifications of the explanations given by the physicians. As a result, we have modified the combination rules only slightly.

6. Lessons learned

From the medical point of view, ICONS offers advice in a complex knowledge domain, especially for the less-experienced physician. ICONS could help to make decisions for antibiotic therapy more stable and safe, independent of the knowledge of the physicians on duty. The system also considers local resources and conditions so that calculated antibiotics can be performed more precisely and valuable if site-specific information is provided. Concerning the correctness of ICONS advice, we could state that it performs comparatively well against other similar systems [36]. In addition, the program offers prices of different adequate therapies for the consideration of the economic factor of a treatment regime.

The process of decision making of physicians is very often based on cases they have encountered in their professional life. Therefore, the medical literature and computer-based medical records offer numerous easily accessible case collections. However, the intrinsic medical experience of these case bases is not yet fully used in knowledge-based systems. In our opinion, a suitable technique—CBR—has reached the state of maturity for medical applications [9]. Further fields of applications using the methods employed in ICONS are under development.

To implement such a system at different medical sites needs an adaptation, automatically or manual of site-specific background knowledge (antibiotics, resistance, buildings, locations of wards, OP theatres, etc). CBR, however, provides at least a site-specific knowledge base of cases in contrast to knowledge-based system using other architectures (e.g. rules). This decreases the effort to adapt a knowledge base to the local situation, essentially.

The results of the evaluation by questionnaire can be summarized as follows:

1) The general idea of ICONS, namely to determine calculated pathogen spectra concerning the infected organ systems and subsequently to attempt to cover the current patient’s spectrum, was approved by all physicians at the two clinics.
2) Concerning the correctness and the quality of ICONS’s recommendations, the opinions of the physicians deviated from each other. Though most physicians rated ICONS’s recommendations as correct, suitable and useful, few of them said that specific therapies should not be recommended. This variability between medical experts does not only occur in assessing clinical situations, but also in the clinical practice. And it is one of the main reasons for the difficulties to develop golden standards.

A domain-specific problem arises at large medical sites where the decision on antibiotics therapy is distributed. In our Medical School for routine cases of infections physicians make the decision on antibiotics on wards and ICUs. The Department of Microbiology monitors the spread of resistance in time and space and the Department of Pharmacology contributes the individual dose based on measured peak and through values of the antibiotic in serum. Therefore, we distributed ICONS: the wards could use RESISTANCE and ICONS and RESIS-3D is for the Department of Microbiology. In the Department of Pharmacology the functionality of ABINI is provided to determine individual doses.

7. Future plans

Using a computer-based antibiotics advisor is indispensable the more immunocompromized patients have to be treated in case of an infection as a consequence of all types of transplantations. Resistance development in clinics worldwide makes the problem of adequate antibiotics therapy more difficult. Analyzing resistance courses in space and time on-line allows the tracking of agents and provides the base for clinic-wide treatment recommendations. Therefore, we actually work on generalizing our virtual reality system RESIS-3D.

But there remains an unsolved problem. If a clinic is confronted with a problem pathogen, which is unknown at the clinic or cannot be easily cultured, then new biotechnological methods could be helpful. Microarray systems could obtain signatures of the gene expression of pathogens given the chip contains the sequence of this pathogen. New emerging infectious agents could be tracked worldwide. In the framework of a proteomics project, we are going to build an interface to a microarray gene-expression database. Further pharmacogenomic research based on gene expression data of patients could reveal the dose necessary or unexpected side effects for each individual patient. In future antibiotics advice systems all these functions should be integrated.

References

154


