Abductive Logic Programming in the Clinical Management of HIV/AIDS

Oliver Ray[1] and **Athos Antoniades**[1] and **Antonis Kakas**[1] and **Ioannis Demetriades**[2]

Abstract. This paper presents a new Abductive Logic Programming (ALP) approach for assisting clinicians in the selection of antiretroviral drugs for patients infected with Human Immunodeficiency Virus (HIV). The approach is comparable to laboratory genotypic resistance testing in that it aims to determine which viral mutations a patient is carrying and predict which drugs they are most likely resistant to. But, instead of genetically analysing samples of the virus taken from patients – which is not always practicable – our approach infers likely mutations using the patient's full clinical history and a model of drug resistance maintained by a leading HIV research agency. Unlike previous applications of abduction, our approach does not attempt to find the "best" explanations, as we can never be absolutely sure which mutations a patient is carrying. Rather, the intrinsic uncertainty of this domain means that multiple alternative explanations are inevitable and we must seek ways to extract useful information from them. The computational and pragmatic issues raised by this approach have led us to develop a new ALP methodology for handling numerous explanations and for drawing predictions with associated levels of confidence. We present our *in-Silico Sequencing System* (iS3) for reasoning about HIV drug resistance as a concrete example of this approach.

1 Introduction

The World Health Organisation estimates 40 million people worldwide are infected with Human Immunodeficiency Virus (HIV). Since there is no cure for this disease, medical intervention aims to postpone the life-threatening collapse of immune function known as Acquired Immune Deficiency Syndrome (AIDS) [15]. Although potent combinations of drugs can slow its progression, the treatment of HIV is complicated by the propensity of the virus to accrue mutations that confer resistance to known medications. Thus clinicians are often faced with the task of designing *salvage therapies* for patients whose first- or second-line treatments are failing.

To improve the likelihood of finding effective salvage therapies medical guidelines advocate the use of laboratory *resistance tests* to help identify which viral mutations a patient is carrying and predict which drugs they are most likely resistant to [8]. But such tests require expert interpretation as they cannot reliably detect *minority* strains of the virus in a patient's bloodstream or *archived* strains residing in less accessible tissues which may conceal drug resistant mutations. Thus clinicians must carefully analyse a patient's medical history in order to infer the presence of other likely mutations not revealed by resistance testing.

[1] University of Cyprus, Cyprus, email: {oliver,athos,antonis}@cs.ucy.ac.cy
[2] Gregoriou AIDS Clinic, Cyprus, email: idemetriades@mphs.moh.gov.cy

This paper presents an experimental logic-based approach that can assist in the interpretation of resistance tests by also reasoning with clinical data summarising the success or failure of previous treatments. In particular, we describe a system called *in-Silico Sequencing System* (iS3) that uses Abductive Logic Programming (ALP) [12] to compute sets of mutations that explain a patient's treatment history. A unique feature of this system is that it can make recommendations even in the absence of resistance tests – which may not be an option for many patients due to prohibitive costs, restricted access or technical limitations. In brief, iS3 uses the same rules as state-of-the-art HIV resistance test interpretation algorithms, but, instead of using them *forwards* to predict likely drug resistances implied by observed mutations, it uses them *backwards* to explain the observed drug response of a patient in terms of likely mutations.

A notable feature of this application domain is that even when resistance tests are available we can never be absolutely sure which mutations a patient is carrying, and so multiple explanations are unavoidable. Thus, in contrast to existing work on abduction, which is aimed at finding the so-called "best" explanations, we must adopt a more pragmatic approach that accepts the plurality of solutions and seeks to extract useful information from them. To this end we propose a three phase method comprising: (i) the development of a heuristic ALP model based on experimentally determined domain rules and prior knowledge, (ii) the introduction of general purpose and domain specific preference criteria to enable the extraction of ranked abductive hypotheses from the multitude of explanations, and (iii) the use of test data to validate the model and to calibrate the rankings by associating a level of confidence with each hypothesis.

The computational challenge of generating and handling large numbers of abductive explanations is considerable as thousands of explanations are typically generated for a single patient. In fact, such computational considerations required us to develop a new abductive engine, described in [16], simply to return these explanations in a feasible time. But, the sheer number of explanations is still too great to allow the efficient extraction of useful information. This motivates the introduction of minimality criteria to reduce the number of redundant explanations considered. Thus, we also describe an efficient method of computing minimal abductive explanations by explaining the observed data incrementally according to the temporal sequence by which it is generated. To demonstrate the utility of our approach, this paper introduces the iS3 drug resistance system as a concrete instance of our general ALP methodology.

The paper is structured as follows. Section 2 introduces the background material for HIV/AIDS and ALP. Sections 3 and 4 describe the ALP model and iS3 approach. Section 5 evaluates the approach using the clinical histories and genotypic data from 10 HIV-infected patients. We conclude with a discussion of related and future work.

2 Background

2.1 HIV resistance testing

Since its detection in the 1980's, HIV has become a global epidemic. Although several anti-HIV drugs have been developed, they are not very effective in isolation and so infected patients are usually given combination therapies of 3 or 4 drugs to disrupt different stages in the viral life-cycle [15]. Unfortunately, treatment is complicated by the tendency of HIV to develop mutations that confer resistance to these medications [8]. In essence, mutations are copying errors arising in the viral genome that result in amino acid substitutions in the protein sequences coded by those genes. Such changes can reduce the efficacy of certain drugs which target those proteins. For this reason, *resistance tests* are used to help identify which mutations a patient is carrying and predict which drugs they are most likely resistant to. Such tests are of two types [8]: *phenotypic assays* measure the ability of laboratory-cultured clinical isolates to thrive under various concentrations of a particular drug; while *genotypic assays* employ genetic probing or sequencing technologies to identify mutations in the genomes of viruses from infected patients and then apply statistical or rule-based methods to predict levels of drug resistance (based on data from large clinical association studies). However, resistance tests suffer from three limitations [8]. First, they are expensive (250-750$) and require equipment to which many HIV-infected individuals do not have access. Second, they are ineffective for patients with low blood viral loads ($\leq$ 500-900 copies/ml). Third, they are insensitive to minority strains (comprising $\leq$ 10-20% of the viral population) that could lead to the failure of a salvage treatment. Consequently, when they are available, resistance tests require expert interpretation that also takes into account prior clinical history, and, when they are not available, it would be highly desirable to have a means of inferring likely mutations from clinical history alone. To this end, we propose the use of ALP, which is described below.

2.2 ALP Knowledge Representation

Abduction is a form of inference that reasons from effects to causes. Given a goal or observations G and a (logical) theory T, abduction returns explanations Δ that logically entail the goal with respect to the theory. Each explanation is a set of ground atoms whose predicates are specified as *abducible*. Intuitively, the abducibles A are predicates whose extents are incompletely specified by the theory, but which may be subject to further integrity constraints IC.

For computational purposes we use the framework of Abductive Logic Programming (ALP) [12, 6]. Formally, an ALP theory is a triple (T, A, IC) where T is a logic program (the theory), A is a set of predicates (the abducibles) and IC is a set of logical formulae (the integrity constraints). A goal G is a set of literals and an *explanation* of G with respect to (T, A, IC) is a set Δ of ground atoms with predicates in A such that: $T \cup \Delta \models_{LP} G$ and $T \cup \Delta \models_{LP} IC$, where $\models_{LP}$ is some standard entailment relation of logic programming (in this paper we assume the *stable model* semantics).

Each abductive explanation Δ is a hypothesis that together with the model described in T explains how an experimental observable G could hold. The role of the integrity constraints IC, which are modularly stated in the theory, is to augment the basic model in T with any partial information on the abducible predicates or to impose additional validity requirements on the hypotheses Δ.

3 Model of HIV Drug Resistance

Our drug resistance model is automatically extracted from genotypic interpretation rules maintained by the French national AIDS research agency (ANRS) [3]. These rules come from expert analysis of large clinical trials investigating which mutations are associated with resistance to which drugs. We now briefly explain how these associations are represented in our model using the drug *zidovudine* as an example. The ANRS database contains four rules for this drug:

```
Z1: Mutation Q151M.
Z2: Insertion at codon 69.
Z3: Mutation T215Y/F.
Z4: At least 3 mutations in: M41L,D67N,K70R,
    L210W,T215A/C/D/E/G/H/I/L/N/S/V,K219Q/E.
```

Z_1 states that mutation `Q151M` causes *zidovudine* resistance. The notation `Q151M` refers to a substitution at codon `151` in the virus where the (wild type) amino acid glutamine, `Q`, is replaced by (the mutant) methionine, `M`. In our model, we use atoms of the form `resistant(P,T,D)` to denote that patient `P` is resistant to drug `D` at time `T`, and atoms of the form `mutation(P,T,M)` to denote that patient `P` is carrying mutation `M` at time `T`. Thus we represent Z_1 above by the clause T_1 below. For brevity, we omit the wild types in our formalism – as these are implied by the reference strain – and represent `Q151M` as `151M`. Z_2 states the *insertion* of an amino acid at codon 69 is another indicator of resistance. This is modelled in clause T_2 using "`i`" to denote an insertion. Z_3 states that either `215Y` or `215F` will cause resistance. These *alternate* mutations are represented by the term "`215YF`" in T_3. Z_4 states that resistance will result if the respective mutations are present at any three of the six codons listed. This is encoded in clause T_4 using the predicate `present/4` defined in clauses T_5, T_6 and T_7 to denote the presence in patient `P` at time `T` of at least `N` mutations from the given list. For convenience, the clauses below are all written in standard Prolog notation where variables begin with uppercase letters, the outer universal quantifiers are omitted and commas implicitly denote conjunctions.

```
T1: resistant(P,T,zidovudine) ←
      mutation(P,T,"151M").
T2: resistant(P,T,zidovudine) ←
      mutation(P,T,"69i").
T3: resistant(P,T,zidovudine) ←
      mutation(P,T,"215YF").
T4: resistant(P,T,zidovudine) ←
      present(P,T,3,["41L","67N","70R",
      "210W","215ACDEGHILNSV","219QE"]).
T5: present(P,T,N,[M|Ms]) ← present(P,T,N,Ms).
T6: present(P,T,N,[M|Ms]) ← mutation(P,T,M),
      K is N-1, present(P,T,K,Ms).
T7: present(P,T,0,Ms).
```

The ANRS database also contains more complex rules, built from conjunctions of simpler expressions. An example is the rule V_1 for *indinavir*, which is represented in our model by the clause T_8 below.

```
V1: L90M and at least 2 mutations in:K20M/R,L24
    I,V32I,M36I,I54V/L/M/T,A71V/T,G73S/A,V77I.

T8: resistant(P,T,indinavir) ←
    mutation(P,T,"90M"),present(P,T,2,["20MR","24
    I","32I","36I","54VLMT","71VT","73SA","77I"]).
```

For convenience, iS3 automatically downloads the ANRS rules from the Stanford HIV Drug Resistance Database (HIVDB) [19], where they are stored in a more convenient XML format called the Algorithm Specification Interface (ASI) [2]. The XML is parsed by a *document object model* compiler and processed by a *definite clause grammar* in order to extract a logical theory. In this way, we obtain 64 rules for 16 drugs. Currently, we only process rules with in the database which have the highest confidence rating, as these represent the surest indicators of resistance.

Where possible, alternate mutations like `54VLMT` are modelled by single terms as this reduces the number of essentially equivalent explanations that need be computed. But, to ensure completeness, it is necessary "break up" alternate mutations that partially overlap with other mutations for the same codon. For example, there are three mutations in the database for codon 54, namely, `54VLMT`, `54V` and `54AMV`, which must be decomposed into the non-overlapping fragments `54V`, `54LT`, `54M` and `54A`. For efficiency, alternate mutations are only broken down into the largest possible fragments.

In order to facilitate the computation of minimal abductive explanations some additional equivalence-preserving transformations are applied to the extracted rules such as combining the three rules T_1-T_3 into one single rule `resistant(P,T,zidovudine)` ← `present(P,T,1,["151M","69i","215YF"])`. As described in the next section, this rewriting is useful because the handling of the predicate `present/4` can be optimised in such a way that reduces the number of non-minimal explanations returned by the abductive proof procedure.

The post-processed theory can be used in two different ways. It can either be used *deductively* to perform standard genotypic test interpretation by asserting known mutations like `mutation(p056,2,"215YF")` and deriving implied resistances like `resistant(p056,2,zidovudine)`. More importantly, it can be used *abductively*, for what we call *in-silico* sequencing, by explaining observed resistances in terms of possible mutations. For example, declaring `mutation/3` as the only abducible predicate, there are 23 possible explanations for zidovudine resistance.

This model can now be used to explain the observed ineffectiveness (resp. effectiveness) of a prescribed *set* of drugs by hypothesising the presence (resp. absence) of mutations that imply resistance (resp. non-resistance) to at least one drug in that set. For example, there are 73 explanations for the ineffectiveness of the combination therapy consisting of the three drugs `zidovudine`, `lamivudine` and `indinavir`. Conversely, the effectiveness of a therapy can be explained by the absence of any mutations that would imply resistance to those drugs.

To capture the dynamics of drug resistance, we need an integrity constraint to enforce a key biological assumption in this work: namely, the persistence of mutations. This assumption is motivated by research showing that drug resistant strains of HIV can persist at undetectable levels for long periods (close to the life expectancy of a patient) only to re-emerge when more favourable drugs are prescribed. This constraint, formalised in I_1 below, states that if a patient `P` carries a mutation `M` at time `T1`, then he or she will carry that mutation at all later times `T2`.

I_1: `false` ← `mutation(P,T1,M), T2>T1, not mutation(P,T2,M).`

In summary, our ALP model consists of the rules extracted from the ANRS database, the integrity constraint above and the abducible predicate `mutation/3`. In the next section, we explain how this model is used in iS3 to reason with real patient clinical data.

4 Analysis of Clinical Data

Given a patient's clinical history in the form of facts stating which combinations of drugs were effective or ineffective at given times, iS3 deploys an ALP procedure [16] to infer sets of mutations that explain these observations according to the model in the previous section. The objective is to obtain a set of hypotheses representing different ways of explaining the data which can then be analysed in order derive useful predictions about drug resistance.

As illustrated above, ineffective treatments suggest that certain mutations may have occurred, while effective ones suggest that certain others did not – possibly leading to a violation of integrity that results in the rejection of incorrect explanations. But, for a typical patient – by which we mean one taking 3 different drugs at each of 10 different time points – the computation produces more solutions (many thousands) than can be conveniently analysed.

For this reason, it is necessary to use additional preference criteria to reduce the search space. Assuming that patients adhere to their medications and are regularly monitored to ensure early detection of treatment failure, it is plausible [8] that mutations accrue relatively slowly. We can model this through the notion of *minimality* [12] – which means we only accept explanations from which no smaller explanations can be obtained by removing atoms.

To enforce the minimality of explanations, we optimised the ALP procedure's handling of `present/4` so that, for example, given the subgoal `present(P,T,1,["151M","69i","215YF"])` it does not consider the first two redundant choices if it has previously established `mutation(P,T,"215YF")`. We also explicitly remove any non-minimal explanations from the solution set obtained after processing each time point. In this way, the number of explanations for the typical patient is cut down to a few hundred.

From this set of explanations explaining a patient's clinical history by the mutations they may or may not be carrying, our task is to analyse these explanations in order to predict for which drugs the patient may be resistant. As a first step, we propose (i) to rank each mutation in the list according to the number and size of explanations it appears in and (ii) to rank each drug in our model according to the number of explanations that imply its resistance.

For task (i), we compute a score for each mutation such that each explanation in which mutation appears contributes a fixed value that decreases with the size of the explanation. (We view this score as simple scalar quantity whose only purpose is to allow the ranking of hypotheses.) The clinical history of a typical patient results in a ranking that contains 30 mutations (out of a possible 60 for which we have rules – or rather 100 if compound mutations are broken up) with scores ranging from 0 to 500, where higher scores indicate a greater likelihood the mutation is present. For example, the top ten ranked mutations for patient `p056` are as follows: `215YF` (270), `69i` (270), `151M` (270), `82S` (256), `84AV` (256), `82T` (256), `82AF` (256), `46IL` (256), `90M` (192), `82M` (179), etc.

For task (ii), we consider each drug in turn and then count the number of explanations which imply resistance to that drug by essentially using the model in a deductive fashion to perform standard genotypic interpretation on each hypothised set of mutations. Once again the resulting scores typically range from 0 to 500, with higher scores denoting a greater likelihood of resistance to that drug. For example, the top ten ranked drugs for patient `p056` are as follows: `efavirenz` (252), `stavudine` (240), `zidovudine` (230), `nevirapine` (224), `indinavir` (81), `lamivudine` (58), `emtricitabine` (58), `saquinavir` (55), `ritonavir` (53), `nelfinavir` (49), etc.

5 Calibration and Evaluation

So far we have described a system that, given a patient's clinical history, produces a ranked list of the mutations the patient might be harbouring and the drugs to which they may be resistant. Now we would like to establish some measure of confidence in those lists and identify clinically relevant cut-offs. To do this, we apply the system to data from 10 HIV-infected patients who, over a period of two years, underwent regular medical examinations, routine blood tests, and were genotyped once.[3] Some statistics for these patients are shown in the table below. On average, there are six time points for each patient (abstracted in simple way from his treatment history), of which roughly 2 are treatment failures. For reference, the numbers of minimal explanations and ranked mutations are also shown.

patient ID	time points	treatmt. failures	minimal explns.	ranked mutations
p027	9	3	1283	29
p032	5	2	51	32
p056	7	3	333	40
p085	6	3	138	26
p089	9	3	1628	26
p101	7	2	191	32
p113	6	2	1938	38
p114	5	1	73	28
p166	4	1	73	28
p184	6	2	72	27

The only way of assessing whether the mutations predicted by the system are biologically correct is to compare the predicted mutations with the result of the genotype. But this is problematic given that our system is an attempt to overcome the fact genotyping cannot reveal all of the mutations carried by a patient. Thus, while we accept that any mutations detected by the genotype are carried by the patient, we expect that our system will return mutations that are not detected by the genotype but are carried by the patient nonetheless.

Figure 1 (a) plots the percentage of mutations detected by the genotype (on the y-axis) that are contained in the corresponding percentage of top-ranked mutations (on the x-axis). This graph shows that the observed mutations are distributed evenly through the ranking.[4] In order to obtain a more useful estimate of the system's performance, it is necessary to compensate for selectiveness of the genotype – which is due to the fact that the drugs taken by the patient at the time of genotype reduce non-resistant strains to undetectable levels, while providing the conditions for resistant strains to thrive.

We therefore expect that mutations which are resistant to these drugs will be detected by the genotype with more certainty than those which are not. To account for this, we use our resistance rules to find those mutations that confer no resistance to any of the drugs being taken at the time of the genotype, and we place these mutations further down the ranking while preserving their relative order.

Figure 1 (b) shows the corresponding plot for the adjusted ranking. We see that observed mutations are now ranked much higher, with the majority contained in the top two thirds. While there is insufficient data to draw any significant conclusion[5], this increases our confidence the system is working correctly, and it suggests a useful clinical cut-off in the sense that the top 2/3 of the ranked mutations appear to contain most of the useful information.

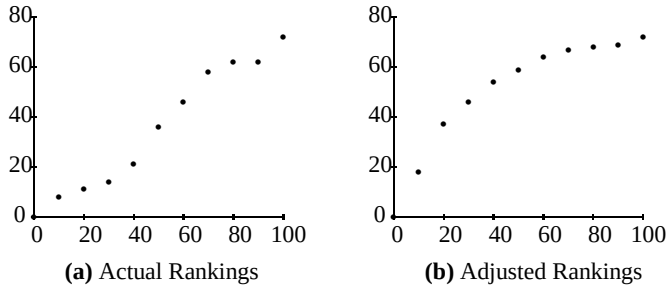

Figure 1. % of genotyped mutations (y) vs. % of top ranked mutations (x)

We would now like to establish whether this 2/3 cut-off also applies to the drug resistances predicted by the system. This is arguably more important since it represents the actual mode of operation for which the system is intended. To do this using the data available to us, we adopted a "leave-one-out" strategy, whereby the system was run on the clinical history of each patient up to *but not including* the last time point with a known (effective or ineffective) outcome at which a new set of drugs was prescribed. We compared the observed effectiveness of the drugs at the next time point with the predicted effectiveness returned by the system.

The results are shown in the table below. For each patient, the 16 drugs are each denoted by one of the symbols '`.`' or '`+`' or '`*`' which are distributed in order of decreasing predicted resistance from left to right. A plus denotes a drug contained in the treatment at next time point. Because the prescribed treatments were all found to be effective at the following time point, we would like for the system to give these drugs a low resistance. As shown in the table, they were mostly predicted medium or low resistance, indicating a better than average performance. A star denotes a drug for which resistance can be inferred using the rules of the system and the mutations in the genotype. We would like the system to rank these highly. As shown below, this is generally the case, once again suggesting a better than average performance. All other drugs are denoted by a dot.

Interestingly, half of the patients were prescribed one drug with high predicted resistance. But, in all cases, the other drugs were ranked much lower – which is consistent with the overall success of the treatment. The system performed very well for `p166`, where resistance to the two top-ranked drugs was independently supported by the genotypic evidence, and two of the drugs found to be effective at the next time point received a low or medium prediction.

Patient ID	Predicted Drug Resistance		
	high	medium	low
p027	`.....`	`..++.`	`...+..`
p032	`.+.**`	`+.*..`	`...+..`
p056	`.....`	`+..+*`	`+...+.`
p085	`+....`	`.+...`	`..+..+`
p089	`+....`	`.+...`	`...+..`
p101	`.+...`	`..+.+`	`+.....`
p113	`.....`	`....+`	`..++..`
p114	`.....`	`.+.+.`	`...+.*`
p166	`**.+.`	`...+.`	`..+...`
p184	`..+.+`	`.+...`	`......`

`+` drugs used in effective therapy at next time point
`*` drugs inferred to be ineffective from genotype data

[3] Unfortunately, clinical data from 10 additional patients was considered too unreliable by an HIV specialist to include in this study.

[4] The graph only rises to 80% as some genotyped mutations are not ranked. These could be given a default score of 0, but this serves no real purpose.

[5] The genotype only returns results for 33 of 97 mutations in our rules, so that the presence or absence of most ranked mutations is unknown.

6 Discussion and Related Work

Previous work on HIV resistance is aimed at predicting drug resistance from the results of genotypic or phenotypic tests performed on clinical isolates from infected patients [1]. Our work does just the opposite, by explaining previously observed resistance in terms of underlying mutations and using these explanations to predict possible drug resistance. Our model of HIV drug resistance is derived from the ANRS genotypic interpretation algorithm [3]. Other rule-based interpretation algorithms could have been used instead, such as those compared in [23] or those available from the Stanford HIVDB web site [19], but ANRS was selected because it was more amenable to the extraction of logical rules.

Previous applications of abduction, such as natural language understanding [9], planning [20, 11], perception [21] and diagnosis [4, 17, 7], take the approach of inference to the "best explanation" [10] by using domain specific constraints to identify one or more optimal hypotheses. The (often implicit) assumption in these applications is that we have sufficient information to discriminate between alternative explanations. Some applications such as [13] and [5] even combine abduction and constraint solving in order to solve the implicit optimisation problems underlying these domains. When this assumption does not hold it is suggested that we carry out "crucial experiments" [18] to obtain further information that would help home-in on the optimal hypotheses.

Our work offers an alternative approach to real-life applications in which the underlying model does not (and cannot) provide enough information to reliably discriminate between competing hypotheses. In these cases, we suggest the emphasis should be placed on extracting useful information from the multitude of explanations and helping the user to make informed decisions. This view is particularly relevant in applications of abduction to scientific theory formation, such as [24, 14, 22], where the task is to improve a necessarily incomplete model of the problem domain.

7 Conclusions and Future Work

This paper introduced a novel AI approach for abductively inferring mutations carried by HIV-infected patients and predicting likely drug resistance from clinical data. The proposed approach is a concrete instance of a new ALP methodology for addressing the issue of multiple explanations through a process of ranking abductive hypotheses extracted from the multitude of explanations and empirically calibrating the system to obtain a measure of confidence in its results. Our in-silico sequencing system, iS3, is a practical implementation with the potential to assist clinicians in the selection of anti-retroviral drugs – especially in cases where laboratory resistance tests are not available. The system automatically keeps itself updated with the latest drug resistance data provided by a leading HIV research body.

We are currently preparing the system for a trial deployment in an HIV/AIDS clinic where we hope to assess the clinical utility of the system and carry out further tests on another cohort of HIV carriers from a different geographical areas. We are also investigating possible improvements to the system by (i) extending the domain model with additional information about common *pathways* by which some viral mutations are known to occur, by (ii) incorporating information about *antagonistic mutations* that partially reverse the resistance effects of some other mutations, and by (iii) exploiting statistical information associated with the drug resistance rules used in the model. We are also keen to examine how our general ALP approach for handling multiple explanations can be applied to other applications.

ACKNOWLEDGEMENTS

We are grateful to Leondios Kostrikis, who helped with the interpretation of the genotypic tests, and to Bradely Betts for his assistance with the Stanford Algorithm Specification Interface. We also thank the anonymous referees for their helpful suggestions.

REFERENCES

[1] N. Beerenwinkel et al., 'Computational methods for the design of effective therapies against drug resistant HIV strains', *Bioinformatics*, **21**(21), 43943–3950, (2005).
[2] B.J. Betts and R.W. Shafer, 'Algorithm Specification Interface for Human Immunodefiency Virus Type 1 Genotypic Interpretation', *journal of clinical microbiology*, **41**(6), 2792–2794, (2003).
[3] F. Brun-Vezinet, B. Masquelier, et al. Agence Nationale de Recherche contre le Sida - AC11 resistance group - genotypic drug resistance interpretation rules (v13), 2005. At http://pugliese.club.fr/index.htm.
[4] L. Console et al., 'Using compiled knowledge to guide and focus abductive diagnosis.', *IEEE Transactions on Knowledge and Data Engineering*, **8**(5), 690–706, (1996).
[5] M. Denecker et al., 'Scheduling by "abductive execution" of a classical logic specification', in *Proceedings of the ERCIM/Compulog workshop on Constraints*, pp. 1–5, (1997).
[6] M. Denecker and A.C. Kakas, *Abduction in Logic Programming*, In Computational Logic: From Logic Programming into the Future, Springer-Verlag, 2001.
[7] J. Gartner et al., 'Psychiatric diagnosis from the viewpoint of computational logic', in *Proceedings of the First International Conference on Computational Logic*, volume 1861 of *Lecture Notes in Computer Science*, pp. 1362–1376. Springer-Verlag, (2000).
[8] M.S. Hirsch et al., 'Antiretroviral Drug Resistance Testing in Adults Infected with Human Immunodeficiency Virus Type 1: 2003 Recommendations of an International AIDS Society-USA Panel', *Clinical Infectious Diseases*, **37**, 113–141, (2003).
[9] J.R. Hobbs et al., 'Interpretation as abduction', *Artificial Intelligence*, **63**(1-2), 69–142, (1993).
[10] J. R. Josephson and S. G. Josephson, *Abductive Inference: Computation, Philosophy, Technology*, Cambridge University Press, 1994.
[11] A.C. Kakas et al., 'ACLP: Abductive Constraint Logic Programming', *Journal of Logic Programming*, **44**(1-3), 129–177, (2000).
[12] A.C. Kakas, R.A. Kowalski, and F. Toni, 'The role of abduction in logic programming', in *Handbook of Logic in AI and Logic Programming*, volume 5, 235–324, O.U.P., (1998).
[13] A.C. Kakas and A. Michael, 'An abductive-based scheduler for air-crew assignment.', *Applied Artificial Intelligence*, **15**(3), 333–360, (2001).
[14] R.D. King et al., 'Functional genomic hypothesis generation and experimentation by a robot scientist', *Nature*, **427**, 247–252, (2004).
[15] Panel on Clinical Practices for Treatment of HIV Infection - US Department of Health and Human Services, *Guidelines for the Use of Antiretroviral Agents in HIV-1-Infected Adults and Adolescents*, 2005.
[16] O. Ray and A.C. Kakas, 'ProLogICA: a practical system for Abductive Logic Programming', in *Proceedings of the 11th International Workshop on Non-monotonic Reasoning*, (to appear 2006).
[17] A. Sattar and R. Goebel, 'On the efficiency of logic-based diagnosis', in *Proceedings of the 3rd international conference on industrial and engineering applications of AI and expert systems*, pp. 23–31, (1990).
[18] A. Sattar and R. Goebel, 'Using crucial literals to select better theories', *Computational Intelligence*, **7**(1), 11–22, (1991).
[19] R.W. Shafer et al. Stanford University HIV Drug Resistance Database, 2005. At http://hivdb.stanford.edu//index.html.
[20] M. Shanahan, 'An abductive event calculus planner', *Logic Programming*, **44**(1-3), 207–239, (2000).
[21] M. Shanahan, 'Perception as abduction: Turning sensor data into meaningful representation', *Cognitive Science*, (to appear 2006).
[22] N. Tran, C. Baral, V. Nagaraj, and L. Joshi, 'Knowledge-based framework for hypothesis formation in biochemical networks', *Bioinformatics*, **21**(supplement 2), 213–219, (2005).
[23] M. Zazzi et al., 'Comparative evaluation of three computerized algorithms for prediction of antiretroviral susceptibility from HIV type 1 genotype', *Antimicrobial Chemotherapy*, **53**(2), 356–360, (2004).
[24] B. Zupan et al., 'Genepath: a system for automated construction of genetic networks from mutant data', *Bioinformatics*, **19**, 383–389, (2003).