ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν Πανεπιστήμιον Αθηνών
ΙΔΡΥΘΕΝ ΤΟ 1837

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
1989 - 2019 30 ΧΡΟΝΙΑ

ΕΠΙΛΕΓΜΕΝΕΣ ΠΤΥΧΙΑΚΕΣ & ΔΙΠΛΩΜΑΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

Τόμος 16
ΑΘΗΝΑ 2019

ΕΠΙΛΕΓΜΕΝΕΣ
ΠΤΥΧΙΑΚΕΣ & ΔΙΠΛΩΜΑΤΙΚΕΣ
ΕΡΓΑΣΙΕΣ

Εκδίδεται μία φορά το χρόνο από το:

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Εθνικόν και Καποδιστριακόν Πανεπιστήμιον Αθηνών,
Πανεπιστημιούπολη, 15784 Αθήνα

Επιμέλεια έκδοσης:
Επιτροπή Ερευνητικών και Αναπτυξιακών Δραστηριοτήτων
Θ. Θεοχάρης (υπεύθυνος έκδοσης), Καθηγητής, Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Η. Μανωλάκος, Καθηγητής, Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Γραφιστική επιμέλεια - Επιμέλεια κειμένων:
Ε. Φλωριάς, ΕΤΕΠ, μήμα Πληροφορικής και Τηλεπικοινωνιών

ISSN 1792-8826

Περιεχόμενα

Πρόλογος

Ο τόμος αυτός περιλαμβάνει περιλήψεις επιλεγμένων διπλωματικών και πτυχιακών εργασιών που εκπονήθηκαν στο Τμήμα Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών κατά το διάστημα **01/01/2018 - 31/12/2018**. Πρόκειται για τον **16ο τόμο** στη σειρά αυτή. Στόχος του θεσμού είναι η ενθάρρυνση της δημιουργικής προσπάθειας και η προβολή των πρωτότυπων εργασιών των φοιτητών του Τμήματος.

Η έκδοση αυτή είναι ψηφιακή και έχει δικό της ISSN. Αναρτάται στην επίσημη ιστοσελίδα του Τμήματος και έτσι, εκτός από τη μείωση της δαπάνης κατά την τρέχουσα περίοδο οικονομικής κρίσης, έχει και μεγαλύτερη προσβασιμότητα. Για το στόχο αυτό, σημαντική ήταν η συμβολή του κ. Ευάγγελου Φλωριά που επιμελήθηκε φέτος την ψηφιακή έκδοση και πέτυχε μια ελκυστική ποιότητα παρουσίασης, ενώ βελτίωσε και την ομοιογένεια των κειμένων.

Η στάθμη των επιλεγμένων εργασιών είναι υψηλή και κάποιες από αυτές έχουν είτε δημοσιευθεί είτε υποβληθεί για δημοσίευση.

Θα θέλαμε να ευχαριστήσουμε τους φοιτητές για το χρόνο που αφιέρωσαν για να παρουσιάσουν τη δουλειά τους στα πλαίσια αυτού του θεσμού και να τους συγχαρούμε για την ποιότητα των εργασιών τους. Ελπίζουμε η διαδικασία αυτή να προσέφερε και στους ίδιους μια εμπειρία που θα τους βοηθήσει στη συνέχεια των σπουδών τους ή της επαγγελματικής τους σταδιοδρομίας.

Η Επιτροπή Ερευνητικών και Αναπτυξιακών Δραστηριοτήτων

Θ. Θεοχάρης (υπεύθυνος έκδοσης), Η. Μανωλάκος

Αθήνα, Ιούνιος 2019

ΠΤΥΧΙΑΚΕΣ ΕΡΓΑΣΙΕΣ

Predicting the Evolution of Communities with Online Inductive Logic Programming

George Athanasopoulos (sdi1300002@di.uoa.gr, geotha1995@hotmail.com)

ABSTRACT

In the recent years research on dynamic social network has increased, which is also due to the availability of data sets from streaming media. Modeling a network's dynamic behavior can be performed at the level of communities, which represent their mesoscale structure. Communities arise as a result of user to user interaction. In the current work we aim to predict the evolution of communities, i.e. to predict their future form. While this problem has been studied in the past as a supervised learning problem with a variety of classifiers, the problem is that the "knowledge" of a classifier is opaque and consequently incomprehensible to a human. Thus, we have employed first order logic, and in particular the event calculus to represent the communities and their evolution. We addressed the problem of predicting the evolution as an online Inductive Logic Programming problem (ILP), where the issue is to learn first order logical clauses that associate evolutionary events, and particular Growth, Shrinkage, Continuation and Dissolution to lower level events. The lower level events are features that represent the structural and temporal characteristics of communities. Experiments have been performed on a real life data set form the Mathematics StackExchange forum, with the OLED framework for ILP. In doing so we have produced clauses that model both short term and long term correlations. An extended version of our work could be found at: https://pergamos.lib.uoa.gr/uoa/dl/frontend/el/browse/2641310.

Keywords: Social Network Analysis, Community Evolution Prediction, Machine Learning, Inductive Logic Programming, Event Calculus, Online Learning

Advisors
Dimitrios Vogiatzis, Collaborating Researcher, NCSR «Demokritos»
George Paliouras, Director of Research, NCSR «Demokritos»
Grigorios Tzortzis, Associate Researcher, NCSR «Demokritos»
Nikos Katzouris, Associate Researcher, NCSR «Demokritos»
Panagiotis Stamatopoulos, Assistant Professor, NKUA

1. INTRODUCTION

A social network is a structure which contains individuals, who are linked to other individuals. The link among them states an interaction which has one or more types of interdependency such as friendship, kinship, common interest, financial exchange. Social networks are often represented as graphs, with nodes representing users and edge representing interactions. Usually a social network changes over time because new individuals join the network, new interactions are developed, or some individuals cease to be active for a short or a long period. This is the predominant behavior especially in streaming social media, such as forums. The social networks are often studied at the level of communities, which represent their meso-scale structure. A group of nodes forms a community if it is densely connected, and sparsely connected to other communities. The said communities are not explicitly formed but rather implicitly as a result of the actions of individual users, that are not random but tend to follow a certain pattern that is related to their similarity to other users. There are many algorithms that have been developed for the detection of communities in networks that are static [6]. In dynamic networks, the communities are influenced over the time by their users' interaction. This influence causes changes in the structure of the communities. Many researchers consider that the structure of a community contains important information for network evolution as a whole. Thus, it is highly imperative to model the dynamic behavior in social networks and try to predict their evolution. In this paper we study the problem of community evolution prediction in dynamic social networks. We address this problem as a supervised learning task where we predict four types of community evolutionary events, *growth, shrinkage, continuation and dissolution*. Various features were investigated in order to understand how they influence the results. Among them, are the structural and temporal characteristics of communities. What is unique in the current approach is that we use a first order logic formalism to represent the correlation between evolutionary events and the input features. Moreover, Inductive Logic Programming (ILP) is used to learn event calculus clauses. Event calculus was chosen because it is human understandable, it can be used to model effect of

actions in time, and the variation we have adopted can perform ILP in an online fashion which is especially useful in streaming media.

2. RELATED WORK

The literature in community evolution prediction is quite extensive in terms of features, classifier types and events predicted. Patil et al. [13] predicted whether a community will disappear or survive. They observed that both the level of member diversity and social activities are critical in maintaining the stability of communities. They also found that certain prolific members play an important role in maintaining the community's stability. Goldberg et al. [8] correlated the lifespan of a community with the structural parameters of its early stages. Brodka et al. [2],[7] tried to predict 6 evolutionary events of communities, i.e. growth, shrinkage, continuation, dissolution, merging and split. They used as features the history of the events of a community in the three preceding timeframes, and the community size in these timeframes. They found that the prediction based on simple input features may be very accurate, while some classifiers are more precise than the others. Kairam et al. [11] tried to understand the factors contributing to the growth and longevity in a social network. They investigated the role that two types of growth (diffusion and non-diffusion) play during a community's formative stages. Diffusion growth is when a community attracts new members through ties to existing members. Non-diffusion growth occurs with individuals with no prior ties to the network. Diakidis et al. [4] studied on-line social networks as a supervised learning task with sequential and non-sequential classifiers. Structural, content and contextual features as well as the previous states of a community are considered as the features that are involved in the task of community evolution. The evolution phenomena they tried to predict are the continuation, shrinking, growth and dissolution. Takaffoli et al. [16] classified the events that may occur in a community as follows: survive:{true, false}, merge:{true, false}, split:{true, false}, size:{expand, shrink}, and cohesion:{cohesive, loose}. First, they tried to predict whether a community will survive, followed by a separate predictor for each of the events. Ilhan et al. [10] proposed a regression ARIMA model to predict values of community features based on the values of the past community instances. Then the predicted community features are used to train a classifier to predict the evolutionary events.

The classifiers proposed in the literature so far, are quite opaque in terms of the model that is learnt. Our approach differs in trying build classifiers based on first order logic, and thus they can be inspected by humans.

3. PROPOSED METHODOLOGY

A dynamic social network is time-stamped, and to be analysed it is segmented into time frames, with an overlap between them to allow for a smooth transition. The problem we are addressing is to predict the form of a community in the next frame, given some features of the existing form of a community. The model that perform the prediction is learnt through Inductive Logic Programming (ILP) and represented as clauses of Event Calculus.

3.1 COMMUNITY FEATURES

In this work, we designed two types of features, the structural and the temporal ones. Structural features represented the physical characteristics of a community such as size, density etc. The temporal features included structural features and evolutionary events that were derived from the past instances of a community, and from relations between past instances of a community.

The instances of the same community at different time frames are considered as one dynamic community, which is formally defined as:

Given a set of time frames 1,2,3,...,T, a dynamic community is a sequence of communities $M = \{C^{k1}_{t1}, ..., C^{kp}_{tp}, , ..., C^{km}_{tm}\}$ such that:

a) $1 \leq t_1 < t_2 < ... < t_m \leq T$
b) $\forall t_i, t_1 < t_i \leq t_m \; \exists \; t_j < t_i, 1 \leq k_j \leq n_{tj}, j=1,...,m : C^{kj}_{tj}$ is considered as the ancestor of C^{ki}_{ti}.

Structural Features

- **Size** is the normalized value for the size of a community in time frame.
- **Density** is the number of community edges over the maximum number of edges, the community could have.
- **Cohesion** is defined as the minimal number of edges in a network that need to be removed to disconnect the community.
- **Ratio Association** is the average internal degree of a community's members.
- **Normalized Association** is the fraction between the double number of intra community connections to the sum of the vertex degrees belonging to the community.
- **Ratio Cut** is the average external degree of a community's members.
- **Normalized Edges Number** is the fraction between the number of intra community connections to the number of edges present in time frame.
- **Average Path Length** shows how close on average two random nodes are.

- **Diameter** is the maximum shortest path between all pairs of nodes in community.
- **Clustering Coefficient**: We set as clustering coefficient of a community, the average of the local clustering coefficient of each node.
- **Centrality measures** how central each node of a community is. We used three centrality measures as features, namely closeness, betweenness and eigenvector centrality.

Temporal features

- **Structural features and Evolutionary events of N ancestors**: One group of temporal features is all the structural features, as described above, as well as the evolutionary events for the first n immediate ancestors of community.
- **Similarity of consecutive communities** is the fraction between the nodes/edges that are common in both instances of the community to total nodes/edges of two instances. The similarity metric between nodes, edges and both, was used.
- **Join nodes ratio** is the percentage of nodes joining a dynamic community as it evolves.
- **Left nodes ratio** is the percentage of nodes leaving a dynamic community as it evolves.
- **Activeness** is the ratio of the number of edges in the current community that also existed in its previous instance, to the number of nodes in current community.
- **Lifespan** is the ratio of the number of time frames between the current community *C* and the very first instance of the same dynamic community (total number of ancestors of *C*), to the maximum number of ancestors could have. The maximum number of ancestors *C* could have is equal to $t_w - 1$, where t_w is the number of the time frame where *C* belongs to. In that case there would be an instance of dynamic community *M* in every time frame from the very first one until time frame $t_w - 1$.
- **Aging of a community**, which is part of the dynamic community *M* is the average age of the community *C* members. The age of a member is increased by 1 every time it is found to be also a member of an ancestor community of *C* in the corresponding dynamic community.

3.2 COMMUNITY EVOLUTION PREDICTION

OLED[12] was used to predict four evolutionary events: *growth, shrinkage, continuation, and dissolution*. Note that OLED handles two-class problems, thus it predicts if a community will sustain or will stop sustaining an evolutionary event. In Figure 1, we present the architecture of the prediction system. The

performance of an ILP system may degrade if the background knowledge provided contains large amounts of irrelevant information so experts are required to set the background knowledge, they believe to be useful. Figure 2 presents an example of background knowledge, where the following types of rules are defined: Rules for community entity recognition; rules for time entity recognition; facts for features' quantized values recognition; rules for values of ground truth recognition; and rules which represent the inertia of Event Calculus. OLED can produce predicates of many forms. For example, an argument of a predicate can be considered as input or as output. Modes declaration is a language that limits the forms of predicates. Figure 3 presents an example of modes. The form of rules that OLED learns is presented in Figure 4. In the head of the rule, predicted_event is one of labels we try to predict (growth, shrinkage, continuation, dissolution). Notice that the $community_i$ and $time_j$ indices are the same in the body and head.

Rules can be interpreted as if $feature_1$ of community $community_i$ has value1 at $time_j$ and the same is true for the rest of features then the initiation of event predicted_event is fired. This means that the predicted_event will start to occur at $time_{j+1}$. The happensAt predicates that are required will be discovered by OLED. Likewise, if the body of rule (2) is true then the termination of event predicted_event is fired, thus the predicted_event will stop to occur at time $time_{j+1}$.

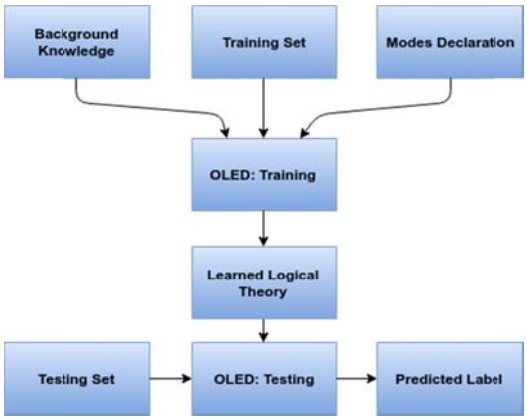


Figure 1: Learning Architecture

Background Knowledge File	
holdsAt(F,T_e) :- fluent(F), holdsAt(F,T_s), not terminatedAt(F,T_s), $T_e = T_s + 1$, time(T_s),time(T_e).	holdsAt(F,T_e) :- fluent(F), initiatedAt(F,T_s), $T_e = T_s + 1$, time(T_s),time(T_e). Inertia of Event Calculus
fluent($growth(X)$) :- community(X).	Ground truth recognition
community(X) :- happensAt(size(X,_),_). community(X) :- happensAt(density(X,_),_).	Community entity recognition
time(X) :- happensAt(size(_,_),X). time(X) :- happensAt(density(_,_),X).	Time entity recognition
value(1..5).	Features' quantized values recognition

Figure 2: Example of A OLED's Background Knowledge File

Mode Declarations File	
modeh(initiatedAt($growth$(+community),+time)) modeh(terminatedAt($growth$(+community),+time))	The form of the rule's head
modeb(happensAt(size(+community,#value),+time)) modeb(happensAt(density(+community,#value),+time))	The form of the rule's body

Figure 3: Example of A OLED's Mode Declarations File

Rules
$\text{initiatedAt/terminatedAt}(< predicted\_event >(< community_i >),< time_j >) :-$ $\text{happensAt}(< feature_1 >(< community_i >,< value_1 >),< time_j >)),$ $\ldots$ $\text{happensAt}(< feature_n >(< community_i >,< value_n >),< time_j >)).$ (1)/(2)

Figure 4: Rules That OLED Learns

Training and Testing As shown in Figure 1, the dataset was split into training and testing sets according to the Time Series Cross Validation, because it takes into account the temporal relationship between the training and testing sets. Training data come from previous time steps and aim to predict the evolution at the next time step.

- **Fold 1**: Training set includes low events of communities from timeframes F_1, F_2 and high events of communities in timeframe F_2. Testing set includes low events of communities from timeframe F_2 and high events of communities from timeframes F_2, F_3.
- **Fold 2**: Training set includes low events of communities from timeframes F_1, F_2, F_3 and high events of communities from timeframe F_2, F_3. Testing set includes low events of communities from timeframe F_3 and high events of communities from timeframes F_3, F_4.
- ... ...
- **Fold T-2**: Training set includes low events of communities from timeframes F_1, F_2, ..., F_{T-1} and high events of communities from timeframe F_2, F_3, ..., F_{T-1}. Testing set includes low events of communities from timeframe F_{T-1} and high events of communities from timeframes F_{T-1}, F_T.

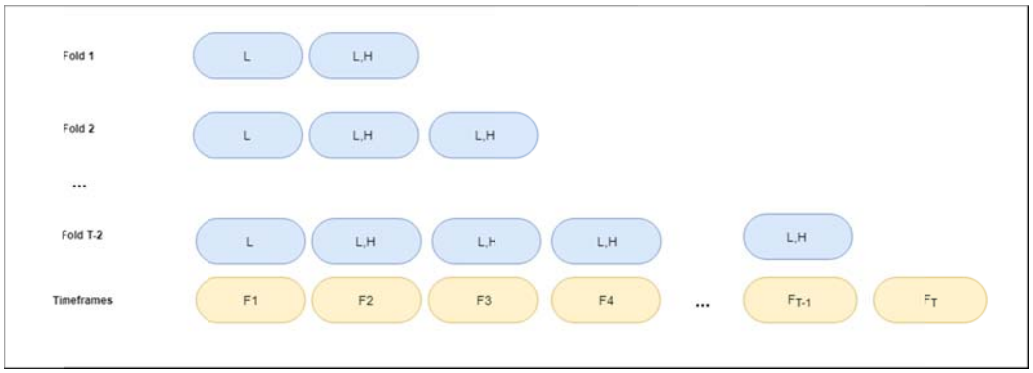


Figure 5: Training Sets using Time Series Cross Validation

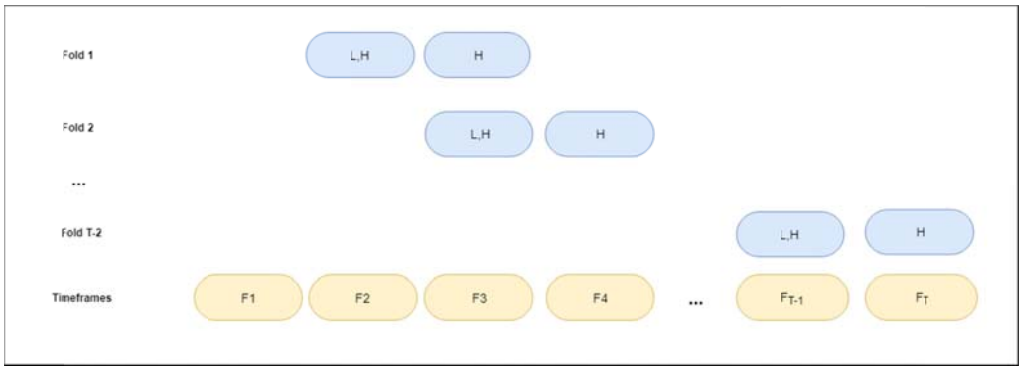


Figure 6: Test Sets using Time Series Cross Validation

T is total number of timeframes in the dataset. Note that the first timeframe has no evolutionary events (high events) since there is no previous timeframe to base the evolution of the first timeframe. Respectively, the last timeframe has not features (low events) because there is no next timeframe to predict evolution of its communities. Also, in the training set we comprise the low level events of the timeframe that we are going to predict so that OLED extracts the time variable for high level events. Finally, notice that in the testing set we also comprise the high level events of the previous timeframe than that we are going to predict. This is required by OLED to initiate the inertia of every community's event.

OLED as an online learner splits its input into chunks. In our experiments, we choose chunks of size 2. Thus, the imported timeframes for the training procedure are split into chunks two by two. We changed the functionality of OLED so that it creates rolling chunks. It means that first chunk contains the timeframes 1,2; the second one the timeframes 2,3; the third the timeframes 3,4 and so on. This is necessary because, for example, timeframe 2 has to be in the first and second chunk. In the first chunk, we need the high level events of timeframe 2 for getting the ground truth. While in the second chunk we use the low level events of timeframe 2 as features for our supervised learning classification.

The outline of training process is the following: Initially there is an empty theory. Each time OLED receives a chunk of training examples and transforms the existing theory to satisfy as close as possible the right prediction of the current examples. When the training process is completed, a logical theory is derived as the learnt model. Its form is illustrated in Figure 4. Using this theory, we predict the evolutionary events of communities which are in testing set.

4. EXPERIMENTS

Dataset description The data were collected from the Mathematics Stack Exchange forum[1], which is a question and answer site for mathematics. All questions are tagged with their subject areas. The dataset comprises 376,030 posts, 261,600 answers and comments, between 28-09-2009 and 31-05-2013. Each user is represented by a node in a graph and there is an edge between two user nodes if one of them posts an answer or a comment on the other user's post. The dataset was split into 10 equally sized, with respect to the number of posts (questions, answers or comments), timeframes with 60% overlap between them.

Building the ground truth means obtaining community labels per time frame, and then obtaining the evolution of each community across time frames. We considered that a group of users belongs in the same community if they post (questions, answers or comments) about the same topic. In particular, we used tags to determine the communities and since on each post there are multiple tags, thus each user will be assigned to multiple communities. Answer and comment posts inherit the tag of the question they correspond to. Also, communities with no more than 3 members were removed. The evolutionary events of each community (Growth, Shrinkage, Continuation, Dissolution) were obtained by thresholding. In particular if the size of the community in the next time frame is more (less) than 30 nodes compared to the size in the current frame then the community grows (shrinks).

There are communities which do not appear in each timeframe, although they may not have been dissolved yet. It happens because communities with few members in a timeframe are pruned from dataset. So, we are looking for the evolution of a community in every timeframe of the dataset and consider a community as dissolved only after its last appearance. The evolutionary events of dataset are imbalanced. In particular, the percentage of each class is: Growth: 0.5%, Shrinkage: 0.2%, Continuation: 90% and Dissolution: 0.3%.

The features were quantized into 5 levels and represented as low level events. The high level events are the evolutionary events. Experiments were executed with both structural and temporal features, where the number of ancestors was set to 4. At the end, the dataset was split in training and testing sets using Time Series Cross Validation method. Because the data are highly imbalanced apart from Micro Average measures, we also used Macro Averages.

---

1 https://archive.org/download/stackexchange

Experiment with all events The results on all events with structural features appear in Figure 7. The macro precision is highest in the dissolution. The dataset with the temporal features contains the features of the previous 4 instances of a community, the first timeframe for this dataset is at time 5 (see Figure 8). The theory which was derived for the dissolution event was empty. The predictor could not evaluate any rule with high score because there were not many available examples, since the number of timeframes (6, from F5 to F10) and the number of communities is small. Thus, the dissolution event is not included in the experiments with temporal features. The growth and shrinkage events with temporal features have lower performance than the best corresponding events with structural features. But for the continuation event, the reverse is true for both micro and macro values.

	Growth	Shrinkage	Continuation	Dissolution
Micro/Macro Precision	0.2358/0.6037	0.1884/0.5832	0.9293/0.8066	0.6512/0.8125
Micro/Macro Recall	0.3027/0.6316	0.1512/0.5671	0.9760/0.6939	0.2629/0.6289
Micro/Macro Fscore	0.2651/0.6174	0.1677/0.5750	0.9521/0.7460	0.3746/0.7090

Figure 7: All events Structural features

	Growth	Shrinkage	Continuation
Micro/Macro Precision	0.1828/0.5730	0.1882/0.5743	0.9182/0.9222
Micro/Macro Recall	0.1828/0.5730	0.1633/0.5649	0.9955/0.6932
Micro/Macro Fscore	0.1828/0.5730	0.1749/0.5696	0.9553/0.7915

Figure 8: All events Temporal features

Experiment with weighted TPs, FPs, FNs. A problem is that the learnt theories contain more termination than initiation rules, thus the initiation of some events does not happen. It means OLED predicts a negative event (i.e. event that does not occur) for a community at next timeframe but in reality, it is a positive event (i.e. the event occurs). In this case the FNs frequency of OLED is increased. The numbers of initiation and termination rules are not balanced because OLED evaluates its rules based on TPs, FPs and FNs values. Using these values, it computes a score which evaluates the accuracy of a rule. To control score's value, we can add weights on TPs, FPs, FNs values during the training. For example, if the FNs weight is set to 10, it means that the FNs will be considered as ten times more than it really is, in other words the termination rules will overestimate the termination condition. Thus, the score of termination rules is getting decreased. With this way we focus more in quality than quantity of termination rules. In Figure 9, we present the best weights for each class in an experiment with structural and temporal features. The results are presented in Figure 10 and Figure 11 for the experiment with structural features and with the temporal features respectively.

	TPs-weight	FPs-weight	FNs-weight
Growth	1/1	1/5	15/1
Shrinkage	20/1	1/1	15/1
Continuation	1/1	1/1	1/15
Dissolution	1	1	15

Figure 9: Best weights for each class with structural/temporal features

	Growth	Shrinkage	Continuation	Dissolution
Micro/Macro Precision	0.2376/0.6055	0.1127/0.5533	0.9247/0.9623	0.8036/0.8878
Micro/Macro Recall	0.3487/0.6519	0.8023/0.8187	0.9845/0.6772	0.2113/0.6047
Micro/Macro Fscore	0.2826/0.6278	0.1977/0.6603	0.9537/0.7950	0.3346/0.7194

Figure 10: Weights on TPs,FPs,FNs - Experiment With Structural Features

	Growth	Shrinkage	Continuation
Micro/Macro Precision	0.2184/0.5913	0.2459/0.6032	0.9182/0.9222
Micro/Macro Recall	0.2043/0.5857	0.1531/0.5654	0.9955/0.6933
Micro/Macro Fscore	0.2111/0.5885	0.1887/0.5837	0.9553/0.7915

Figure 11: Weights on TPs,FPs,FNs - Experiment With Temporal Features

While we were trying various values to weights, we noticed in the results that:

- If TPs's weight is increased then TPs is increased, FPs is increased and FNs is decreased because the number of initiations rules is increased.
- If FPs's weight is increased then TPs is decreased, FPs is decreased and FNs is increased because the number of initiations rules is decreased.
- If FNs's weight is increased then TPs is increased, FPs is increased and FNs is decreased because the number of termination rules is decreased.

We tried to increase the low TPs number by setting appropriate weights, but FPs also increased. OLED overestimated the initiation condition because its initiation rules are not specialized enough to detect correctly in which communities an event will occur. This is a strong indication that with the current features, OLED performance could not improve.

An advantage of OLED is that the predictive model (Theory) it derives is human-readable. Thus, the rules can be read, analyzed and interesting conclusions can be derived from them. Some of the best performing rules are shown in Figure 12. Transferability of the knowledge derived from the rules to new datasets is also an interesting possibility.

Some features appeared more often in rules of specific evolutionary events than others; while some never appeared. In Figure 13 and 14 we present for each evolutionary event (growth, shrinkage, continuation, dissolution), the frequency of the structural features the bodies of rules.

initiatedAt(*growth*(X0),T1) :- happensAt(density(X0,1),T1), happensAt(diameter(X0,2),T1).
terminatedAt(*growth*(X0),T1) :- happensAt(ratio_cut(X0,3),T1), happensAt(average_path_length(X0,3),T1), happensAt(normalized_edges_number(X0,5),T1).
terminatedAt(*growth*(X0),T1) :- happensAt(ratio_cut(X0,3),T1), happensAt(closeness_centrality(X0,3),T1), happensAt(normalized_edges_number(X0,5),T1).
terminatedAt(*growth*(X0),T1) :- happensAt(cohesion(X0,2),T1), happensAt(average_path_length(X0,3),T1), happensAt(diameter(X0,2),T1).
terminatedAt(*shrinkage*(X0),T1) :- happensAt(ratio_association(X0,3),T1).
terminatedAt(*shrinkage*(X0),T1) :- happensAt(average_path_length(X0,2),T1).
terminatedAt(*shrinkage*(X0),T1) :- happensAt(closeness_centrality(X0,2),T1), happensAt(ratio_cut(X0,1),T1).
initiatedAt(*shrinkage*(X0),T1) :- happensAt(eigenvector_centrality(X0,1),T1), happensAt(ratio_association(X0,5),T1).

Figure 12: Rules learnt in the best experiment: Growth and Shrinkage

Growth	Percentage	Shrinkage	Percentage
diameter	17.68%	ratio_association	20%
cohesion	13.26%	ratio_cut	13.55%
ratio_cut	11.60%	cohesion	11.61%
average_path_length	11.60%	clustering_coefficient	10.97%
density	8.29%	eigenvector_centrality	9.03%
ratio_association	7.73%	density	8.39%
clustering_coefficient	7.73%	average_path_length	7.10%
size	6.63%	closeness_centrality	6.45%
closeness_centrality	6.08%	centrality	3.871%
eigenvector_centrality	3.87%	diameter	3.87%
normalized_edges_number	2.76%	betweenness_centrality	2.58%
centrality	2.76%	size	1.29%
		normalized_edges_number	1.29%

Figure 13: Structural features frequency: Growth and Shrinkage

Continuation	Percentage	Dissolution	Percentage
ratio_cut	16.07%	clustering_coefficient	25.93%
ratio_association	15%	cohesion	15.74%
clustering_coefficient	10%	betweenness_centrality	10.19%
density	9.64%	diameter	9.26%
diameter	8.21%	size	8.33%
closeness_centrality	8.21%	normalized_edges_number	6.48%
eigenvector_centrality	7.14%	ratio_association	5.56%
centrality	6.79%	closeness_centrality	3.70%
betweenness_centrality	6.43%	average_path_length	3.70%
normalized_association	4.64%	ratio_cut	2.78%
average_path_length	4.64%	normalized_association	2.78%
normalized_edges_number	2.5%	centrality	2.78%
size	0.71%	density	1.85%
		eigenvector_centrality	0.93%

Figure 14: Structural features frequency: Continuation and Dissolution

5. CONCLUSIONS

We tried to predict the evolution of communities in a dynamic social network. The evolution of a community is described as the occurrence of growth, shrinkage, continuation and dissolution events. We carried out the prediction using OLED, an Inductive Logic Programming system for learning logical theories from data streams. The ground truth was built by considering the tags of the raw data set, and by setting thresholds on the community size. As features we used structural characteristics of communities. Moreover, we have also designed temporal features where a preset number of previous instances of each community were used as well as features that capture change between consecutive instances of a community. Subsequently, the features were quantized. The dataset was obtained from the Mathematics Stack Exchange forum. We presented the micro and macro averages, because the classes (Growth, Shrinkage, Continuation and Dissolution) were unbalanced. We also investigated the best pruning values for the theory in OLED, which did not improve the results. Overall, the experiments with the temporal features had a worse performance than the experiment with the structural features, probably because there were not many timeframes. Then, we execute experiments where OLED learnt rules that represent long range relationships between an evolutionary event and features. Subsequently, weights were applied to TPs, FPs and FNs values to change rules' scores alleviating the problem of the imbalanced classes. This was the experiment with the best results. Finally, we

presented the features that were the most influential for each evolutionary event.

REFERENCES

[1] Hendrik Blockeel, Luc De Raedt, Nico Jacobs, and Bart Demoen. Scaling up inductive logic programming by learning from interpretations. Data Mining and Knowledge Discovery, 3(1):59–93, Mar 1999. URL:https://doi.org/10.1023/A:1009867806624, doi:10.1023/A:1009867806624.

[2] P. Bródka, P. Kazienko, and B. Kołoszczyk. Predicting Group Evolution in the Social Network. ArXiv e-prints, October 2012. arXiv:1210.5161.

[3] L. De Raedt. Logical and Relational Learning. Cognitive Technologies. Springer Berlin Heidelberg, 2008. URL: https://books.google.gr/books?id=FFYIOXvwq7MC.

[4] Georgios Diakidis, Despoina Karna, Dimitris Fasarakis-Hilliard, Dimitrios Vogiatzis, and George Paliouras. Predicting the evolution of communities in social networks. In Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics, WIMS '15, pages 1:1–1:6, New York, NY, USA, 2015. ACM. URL: http://doi.acm.org/ 10.1145/2797115.2797119, doi:10.1145/2797115.2797119.

[5] Opher Etzion and Peter Niblett. Event Processing in Action. Manning Publications Co., Greenwich, CT, USA, 1st edition, 2010.

[6] Santo Fortunato. Community detection in graphs. CoRR, abs/0906.0612, 2009.

[7] Bogdan Gliwa, Piotr Bródka, Anna Zygmunt, Stanislaw Saganowski, Przemyslaw Kazienko, and Jaroslaw Kozlak. Different approaches to community evolution prediction in blogosphere. In Advances in Social Networks Analysis and Mining 2013, ASONAM '13, Niagara, ON, Canada - August 25 - 29, 2013, pages 1291–1298, 2013. http://doi.acm.org/10.1145/2492517.2500231, doi:10.1145/2492517.2500231.

[8] Mark Goldberg, Malik Magdon ismail, Srinivas Nambirajan, and James Thompson. Tracking and predicting evolution of social communities, IEEE International Conference on Social Computing, pp. 780-783, 2011.

[9] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association, 58(301):13–30, 1963. doi:10.1080/01621459.1963.10500830.

[10] Nagehan İlhan and Şule Gündüz Öğüdücü. Predicting community evolution based on time series modeling. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, ASONAM '15, pages 1509–1516, New York, NY, USA, 2015. ACM. URL: http://doi.acm.org/10.1145/2808797.2808913, doi:10.1145/2808797.2808913.

[11] Sanjay Ram Kairam, Dan J. Wang, and Jure Leskovec. The life and death of online groups: Predicting group growth and longevity. In Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, WSDM '12, pages 673–682, New York, NY,USA, 2012. ACM. URL: http://doi.acm.org/10.1145/2124295.2124374, doi:10.1145/2124295.2124374.

[12] Nikos Katzouris, Alexander Artikis, and Georgios Paliouras. Online learning of event definitions. Theory and Practice of Logic Programming, 16(5-6):817–833, 2016. doi:10.1017/S1471068416000260.

[13] Akshay Patil, Juan Liu, and Jie Gao. Predicting group stability in online social networks.In Proceedings of the 22Nd International Conference on World Wide Web, WWW '13, pages 1021–1030, New York, NY, USA, 2013. ACM. URL: http://doi.acm.org/10.1145/2488388.2488477, doi:10.1145/2488388.2488477.

Path Planning for Incline Terrain Using Embodied Artificial Intelligence

Georgios Kamaras (sdi1400058@di.uoa.gr, georgioskamaras@gmail.com)

ABSTRACT

The inclination of the terrain should always be taken into account during path planning. In the case of steep inclines, following a non-shortest, smoothly curved path can substantially increase a mobile platform's ability to operate. In my thesis I presented a path planning method that produces such paths given the terrain inclination, while also taking into consideration the obstacles in the field.

Keywords: Unmanned Ground Vehicle, Offline Path Planning, Incline Terrain Path Planning, Bézier Curves, Steepest Ascent Hill Climbing, Evolutionary Algorithm.

Advisors
Stasinos Konstantopoulos, Associate Researcher (N.C.S.R. "Demokritos"), Panagiotis Stamatopoulos, Assistant Professor

1. INTRODUCTION

Traversing challenging terrain is one of the main goals of field robotics research. The problem of traversing steep inclines has already received considerable research attention regarding the hardware design of wheeled [1, 2], legged [3] and crawling [4] platforms as well as motor control [5-7]. This problem also gives researchers the chance to explore how intelligent behavior, like selecting the suitable path under certain circumstances, can increase a robot's autonomy without modifying its hardware. To the best of my knowledge, at the time of writing this paper, the most sophisticated literature on path planning proves how paths constructed from circular arcs are more energy-efficient than the shortest path for a skid-steered rover traversing loose or slippery terrain [8].

In this paper we are going to explore the novel idea of using smoothly curved Bézier paths to traverse a terrain region of steep inclination. Created by stitching together Bézier curves, these paths avoid sharp turns that stress or exceed skid-steered rovers' capabilities. Simultaneously, they are optimized for a given field configuration, taking into account both its inclination and the obstacles present in it. In the remainder of this paper I first provide the background on optimization and Bézier curves (Section 2) and then proceed to present my method (Section 3), experimental results (Section 4), and conclusions (Section 5).

2. BACKGROUND

In this section we are going to lay the theoretical foundations of our path planning method through a sort presentation of the Optimization Algorithms that produce our paths and the Bézier curves that shape these paths.

2.1 Optimization Algorithms

The area between the robot's current position and its goal position can be seen as a graph of waypoints (vertices) connected with edges when it is possible to move from one to the next, and where these edges are labelled with the cost of moving between these two waypoints. In order to plan a path from its current position to the goal the robot needs to traverse this graph until it reaches the goal, having accumulated the minimal cost from the steps it took.

In order to find solutions in such a search space, we will explore *Steepest Ascent Hill Climbing* and *Evolutionary Algorithms*. As a simple and fast algorithm Steepest Ascent Hill Climbing (S.A.H.C.) is a good baseline approach than can often greedily and quickly find a "good enough" path. However, it may fail in the presence of obstacles, not being able to backtrack from solutions that cross a lethal obstacle. Because of this, we will also include in our investigation its *N-Best* variant where each step maintains the *N* best solutions. This way the algorithm has alternatives to explore in case it reaches a locally optimal solution that crosses an obstacle. However, the N-Best algorithm may also fail to produce a viable path in cases of large obstacles, because it cannot backtrack deep enough to plan a path around the obstacle.

Obstacles are discontinuities in the cost heuristic that S.A.H.C. algorithms do not take into account until they are actually forced to backtrack. So, a natural choice for an algorithm to compare against S.A.H.C. is *Evolutionary Algorithms* that evaluate and iteratively improve complete solutions. Transferring concepts from biological evolution, Evolutionary Algorithms (E.A.) use the cost function to select the best individuals among a "generation", which are used to derive the next generation by randomly crossing over their "chromosomes", in our case the waypoints along a path. To allow exploration outside of local optima, some of the individuals may also get randomly "mutated". E.A.s are computationally demanding as they evaluate a far larger set of candidates than S.A.H.C. algorithms. Nevertheless, summing the costs along a path adds a lower overhead than, for example, evaluating generations in computationally expensive simulations.

2.2 Producing Smooth Curves

According to computer graphics research, given two endpoints and a set of *control points* that outline a path between them, the most prominent approach of

formulating curves that balance between being smooth and passing as closely as possible through all control points are Bézier curves [9]. We will use the *quadratic Bézier curve* which has a single control point and is mathematically defined as follows:

$$C_t = (1-t)^2 P_0 + 2(1-t)tP_c + t^2 P_1,\ t \in [0,1]\ ,$$

where P_0 and P_1 are the endpoints, P_c is the control point, and t is a variable running from 0 to 1. Naturally, $C_0 = P_0$ and $C_1 = P_1$, but it is not necessary that there is any value for t such that $C_t = P_c$.

3. OPTIMIZING CURVE PARAMETERS

In this section we present how we can leverage the theory summarized in Section 2 in order to create smoothly curved paths and we discuss the conditions under which each optimization algorithm is most useful.

3.1 An Outline of our Method

3.1.1 Assumptions

The approach presented in this paper targets a situation where a robot at a start position at the bottom of a steep incline needs to reach a goal at the top of the incline. The full path is decided before the robot starts following it. We assume that the robot is able to estimate the inclination, an ability well-established in the traversability estimation literature [10]. We also assume that the robot is able to perceive from the bottom all obstacles that it needs to avoid and that these obstacles are static. Finally, we assume the existence of a standard navigation system able to implement a path provided as a series of waypoints.

3.1.2 Representation

We diverge from the path planning literature by not searching for an optimal series of waypoints, but rather for an optimal series of curves as we defined them in Subsection 2.2. In this manner, the optimization search space is the space of curve parameters and not the space of waypoints. Once the optimal series of abstract curves has been computed, the method produces from that the waypoints that need to be given to the navigation system. Operating in this search space is more efficient because all candidate paths (Bézier paths) are already possible for skid-steered rovers, not containing unimplementable segments, like sharp turns.

3.1.3 Path Cost

We will sum a cost calculated for any point on the line as follows:

- We want to apply bias towards solutions that minimize path length. So, we include a term that depends on the distance d_x between point x and the straight line connecting the initial position and the goal, normalized to the length of this straight line. Steeper slopes should tolerate greater deviations from the shortest path, so the term is $k_1 d_x / s$, where k_1 is a constant and s is the overall slope.
- We want to apply bias towards solutions that make smooth transitions between curves, minimizing abrupt changes of direction when changing from one curve into the next, and also minimizing the number of zig-zags needed to reach the goal. For each waypoint, this can be expressed as

maximizing the angle formed by connecting three consecutive waypoints, so that it is as close as possible to $180°$. However, this is risky for steeper slopes, so the term expressing this bias is $k_2 s/a_x$ where k_2 is a constant and a_x is the angle formed by connecting x with the two previous waypoints.

- We want to take pitch and roll into account, and we observe that higher pitch values make the path riskier, but higher roll values make it safer. We express this as $k_3(p_x - r_x)$ where k_3 is a constant and p_x, r_x the pitch and roll at x.

We define path cost as the sum of the above terms over all points in a path P:

$$cost(P) = \sum_{x \in P} k_1 \frac{d_x}{s} + k_2 \frac{s}{a_x} + k_3(p_x - r_x)$$

3.2 Optimal Path Generation

As discussed in Section 2, generating a path can be formulated as searching for the optimal traversal of the graph that connects the cells of the grid map that are reachable from each other. The start position is P_0 of the first local Bézier curve. We search the incline area's grid by taking two rows of it at each step. In the S.A.H.C. generator, greedily, we try to place the control point P_c, which is not part of the local path, in the first row and the end point P_1 in the second row. First, we calculate each local Bézier curve's total cost. Then, we examine all the possible local Bézier curves and we select the locally optimal Bézier curve and we add it to the global Bézier path. In the next step we assign the P_1 of the latest local Bézier curve to be the P_0 of the next local Bézier curve and we repeat the same process. When we reach the last row of the grid, we assign P_1 to the goal position. This way we create a globally “good-enough” Bézier path, by combining locally optimal decisions.

We expect S.A.H.C. to be the most time efficient of our three generators, but also the naivest, as it cannot handle the presence of a lethal obstacle in one of the locally optimal solutions. In such a case, it fails. The N-Best generator builds on top of the S.A.H.C. generator. If a step is inadmissible, due to a lethal obstacle, we return to the previous step and we take the next-best choice. We do this until either we succeed to move to an admissible next step or we run out of choices. If the latter happens the N-Best generator fails. N-Best's time efficiency is expected to be close to the S.A.H.C. generator's. However, the N-Best generator is less naive, as it can successfully handle simple lethal obstacles. Still, it fails in the presence of lethal obstacles formations that get into the way of all of the N best local choices.

Our E.A. generator operates as presented in Section 2. Its termination criteria are our search reaching a designated threshold of examined generations, or our search succeeding in producing an admissible path from start to goal, or a designated threshold of consecutive stagnated generations being reached. A valid smooth curved path from start to goal is an individual and each path's waypoints are the individual's chromosomes. We start our search (evolutionary process) with a specific number of random individuals, which we sort based on their fitness and we enter our generator's main loop. In the beginning of each iteration, we select the best-fit individuals for reproduction. We take every possible pair of these individuals and we subject it to crossover, in order to create the individuals of the next generation. We then evaluate the fitness of our new individuals, we take them and their parents, we sort them all based on their fitness and we proceed to check whether they fulfill the termination criteria. If they do not fulfill the termination criteria we proceed to the next generation. If they fulfill these criteria, we select the best individual of the current generation as a solution to our problem and send it to our navigation system.

In Figure 1 we see the logic level results of using the N-Best and E.A. optimization methods for producing a path in order to climb a $45°$ and a $43°$-45 meters long slope respectively. The colored lines represent the various paths that each algorithm considered. Some of them may have been deemed inadmissible because they coincide with an obstacle, others because they are not locally optimal. In each case, the path that our method finally selected is annotated with small yellow spheres. Each one of these yellow spheres is a waypoint of our path. We use the Bézier curve's equation to produce the minor waypoints between the start and end point of each curve. These waypoints denseness is a platform-specific parameter.

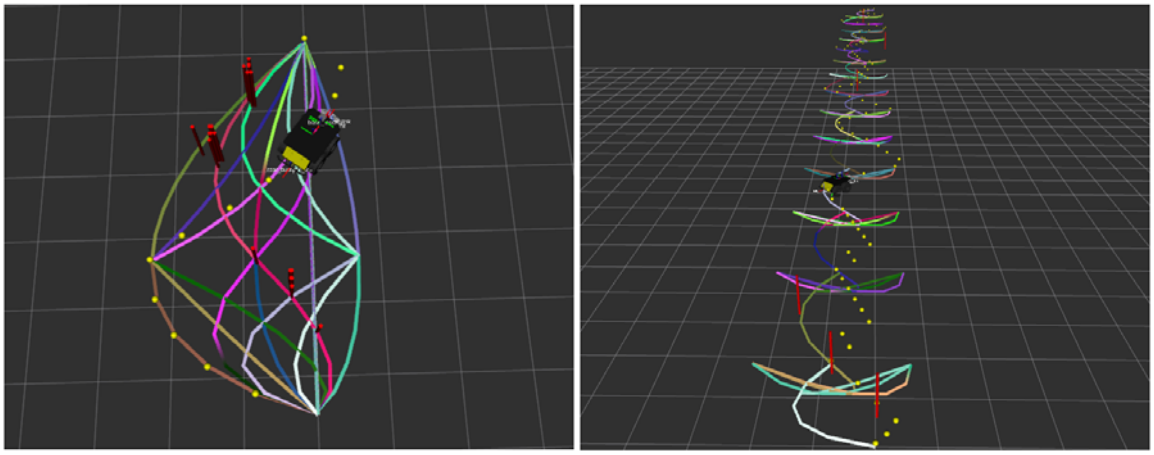


Figure 1: Candidates and selected path (dotted) for a $45°$ slope (left) and a longer $43°$ slope (right).

4. VALIDATION AND DEMONSTRATION

In this section we are going to exam how our path planning method performs in a set of simulation scenarios and we are going to interpret our results.

4.1 Experimental Setup

For the experimental validation of my work I used the Clearpath Husky UGV in simulation and I created three simulation environments. A $35°$ slope environment (Figure 2), a $45°$ slope environment (Figure 2) and a $43°$ slope environment (Figure 3), but with 45 meters of straight-line distance between the start and goal positions.

The $35°$ slope has a reduced (< 1) friction coefficient in order to represent slippery terrain conditions, whereas the $43°$ and $45°$ slopes have a friction coefficient of 1.0, which in real world is considered ideal. The $35°$ slope can demonstrate whether our method can currently help a UGV in a slippery terrain scenario to go further than it would have gone if it was climbing “straight up”. The $45°$ slope was determined experimentally as being ideal for testing my method, as it is steep enough for a wheeled UGV with no special equipment to not be able to traverse it in a straight line path without tipping over, while it is not too steep so that the UGV will not be able to manoeuvre on it. The $43° - 45\ m$ slope environment was created having in mind that the maximum range of the localization sensors available in the market at the time of writing this paper is about 25 meters. It demonstrates that the smooth curves method is viable when the robot cannot see as far as the goal, but somehow has a precise knowledge of its existence and of the challenges that it must face in order to reach it.

For each slope, three scenarios have been created. In the first scenario, the robot will have to reach the goal with no lethal obstacles along the way. In the second, the robot will have to reach the goal while having certain non-complex and small in area obstacles along the way. Finally, in the third scenario, the robot must overcome certain complex and larger in area lethal obstacles along the way.

The search's *resolution* or *search step* is kept equal to *1.5 meters* for all the simulation scenarios. This number derived from experimentation, regarding which resolution is best for our generators to search for a solution in our simulation scenarios, which are characterized by the narrowness of the area in which our robot can move. It is a relatively small resolution considering that our simulations' UGV has a length of *0.3 meters*.

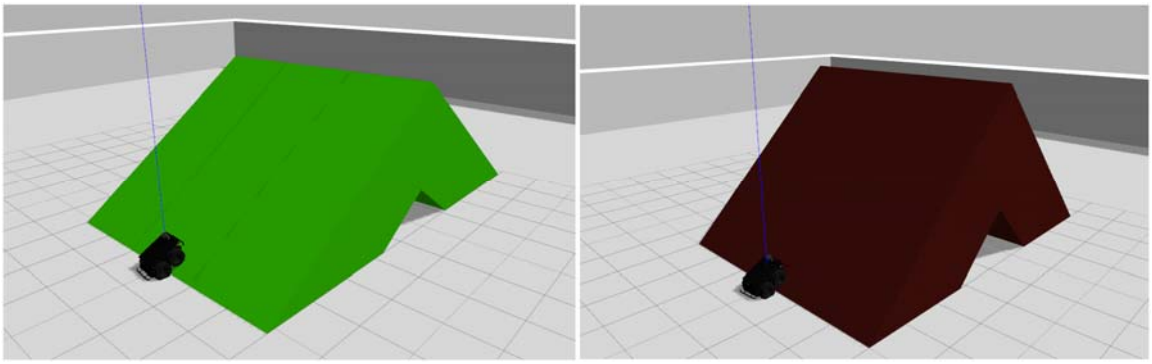


Figure 2: 35° (left) and 45° (right) slopes set-up.

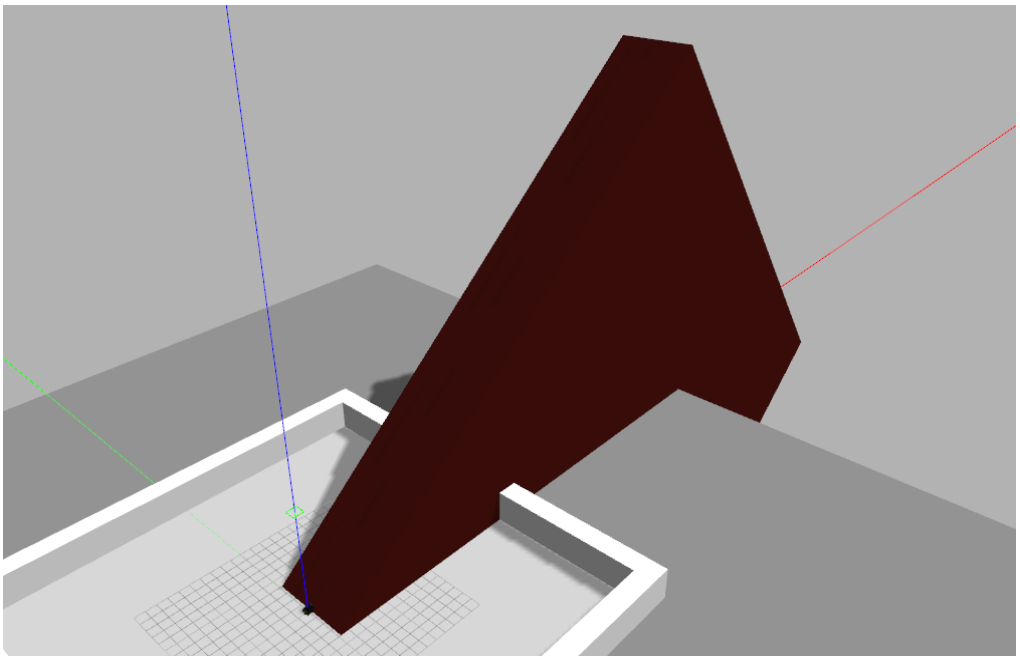


Figure 3: $43^\circ - 45$ meters slope set-up.

4.2 Simulation Results

The robot fails to traverse the 35° simulation environment's slope in a straight up path, as it never reaches its goal due to wheel slippage. The robot fails to traverse the 45° and $43^\circ - 45\ m$ slopes going straight up as it tips over. Now, we will examine and compare the results of simulations, demonstrating our generators capability to help a UGV traverse these slopes. We will judge each simulation's outcome based on its visited states, path size, path length and path cost (Table 1).

Slope and distance	Generator	Visited States	Path Length	$\sum dx$	Cost
$35°, 5.9\ m$	S.A.H.C.	7	$8.03\ m$	$3.68\ m$	6627
	N-Best	7	$8.03\ m$	$3.68\ m$	6627
	E.A.	13	$7.52\ m$	$2.74\ m$	6649
$45°, 5.8\ m$	S.A.H.C.	7	$7.00\ m$	$3.28\ m$	11107
	N-Best	7	$7.00\ m$	$3.28\ m$	10437
	E.A.	208	$6.15\ m$	$2.30\ m$	9575
$43°, 45.0\ m$	S.A.H.C.	85	$57.08\ m$	$17.93\ m$	79481
	N-Best	85	$56.31\ m$	$21.69\ m$	81581
	E.A.	1508	$53.27\ m$	$28.32\ m$	60279

Table 1: Generated path details.

4.2.1 Generated Paths Quality

The robot's effort to climb the $35°$ slope with a S.A.H.C. generated path is not successful compared to climbing “straight up”. While the robot is manoeuvring to follow the curved path's specifications, it loses a substantial part of its progress until that point. As a result, it does not climb as fast or as precisely as it could have by following a straight path from start to goal. From the attempt with an E.A. generated path we verify that the success of the robot at traversing a slippery incline terrain depends on the amount of time that it spends turning and moving away from a straight upward path.

In the $45°$ slope environment, in all cases the robot manages to climb the incline terrain following a relatively smooth path, while also maintaining high stability and showing no wheel slippage. It is also evident that in a logic level the robot manages to escape the obstacles on its way. However, we also see an error accumulating and creating a loss of precision when it comes to the actual obstacle avoidance. This is a localization error created by improper odometry whenever the UGV's front wheels, momentarily lose contact with the terrain.

While reading the E.A. generator's path details in Table 1 we see that we have almost thirty times the visited states compared to the S.A.H.C. generator and the N-Best generator. This is expected because the many crossovers that take place between generations are considered as visiting a new state. The path's length is smaller than the length of our other two generators paths and its cost is considerably improved, which is a great display of the Evolutionary Computation's power when applied to a complex problem. The path's size can be smaller than the S.A.H.C. or N-Best generators or larger, depending on the route that will derive from the evolutionary process.

In each scenario of the $43° - 45\ m$ slope the robot manages to climb the slope with its overall generated path quality prevailing over the accumulating localization error in its effort for ascension. Again, in a logic level, the robot plans its way around any lethal obstacle, however this is not visible while observing the execution of the effort because of the accumulating localization error explained above. As we desired, there are no signs of wheel slippage. For the N-Best and E.A. generated paths the observations that we did in the case of the $45°$ slope scenario, still hold true.

Configuration	Initial Generation Size	Maximum Generations	Similarity	Consistency
Limited	25	50	9/20	45%
Modest	50	100	14/20	70%
Extensive	100	800	18/20	90%

Table 2: E.A. consistency in path production for the $45°$ slope scenario under different search configurations (*Limited*, *Modest* and *Extensive* search).

4.2.2 Evolutionary Algorithm Performance

Being probabilistic by nature, E.A.s may display inconsistency into their output when run with a certain input. However, Tables 2 and 3 demonstrate that a well-designed E.A. can have configurations for which it performs very steadily while being very time efficient. It does so by producing a *Similarly* “viable” path for a high percentage of consecutive runs. In all three configurations, the two *Best Fit* paths of each generation survive to the next one, the evolutionary process is also set to terminate after six *Stagnated Generations* and we consider that we have a stagnated generation if the fittest individual of a generation, which is the path with the least cost in this generation, represents a *Stagnation Rate* of 0.01,

meaning a $\leq 1\%$ improvement compared to the fittest individual of the previous generation.

Slope	Distance	S.A.H.C.	E.A. modest	E.A. extensive
$35°, 5.9\ m$	$5.80\ m$	$0.71\ ms$	$5.33\ ms$	$28.08\ ms$
$45°, 5.8\ m$	$5.71\ m$	$0.88\ ms$	$5.24\ ms$	$29.15\ ms$
$43°, 45.0\ m$	$44.0\ m$	$57.52\ ms$	$20.85\ ms$	$48.73\ ms$

Table 3: Comparison of S.A.H.C.'s and E.A.'s execution times under the same conditions, with no obstacles, for modest and extensive search.

Slope	Distance	N-Best	E.A. modest	E.A. extensive
$35°, 5.9\ m$	$5.80\ m$	$0.70\ ms$	$5.37\ ms$	$27.51\ ms$
$45°, 5.8\ m$	$5.71\ m$	$0.75\ ms$	$5.56\ ms$	$28.48\ ms$
$43°, 45.0\ m$	$44.0\ m$	$31.17\ ms$	$22.17\ ms$	$33.12\ ms$

Table 4: Comparison of N-Best's and E.A.'s execution times under the same conditions, with simple point obstacles, for modest and extensive search.

Slope	Distance	E.A. modest	E.A. extensive
$35°, 5.9\ m$	$5.80\ m$	$6.18\ ms$	$34.12\ ms$
$45°, 5.8\ m$	$5.71\ m$	$6.22\ ms$	$58.38\ ms$
$43°, 45.0\ m$	$44.0\ m$	$29.76\ ms$	$79.47\ ms$

Table 5: E.A.'s execution times, with complex obstacle formations, for modest and extensive search.

4.2.3 Time Efficiency

Reasonably, we run the S.A.H.C. generator considering the absence of any lethal obstacles, the N-Best generator considering the existence of simple obstacles and the E.A. generator considering the existence of complex obstacles. However, the E.A. can also be run with no obstacles and simple obstacles, being our most advanced path generator, it is useful to examine how

it performs under such conditions in comparison to the S.A.H.C. (Table 3) and the N-Best (Table 4). Furthermore, as the E.A. is probabilistic, it is important to examine its performance both for a search of modest intensity and for a more extensive search (Table 5), using the configurations presented in the previous paragraph.

Evidently, the execution time of each path generator is determined by the combination of the straight-line distance between the start and the goal with the presence of lethal obstacles and the algorithm's search configurations. For the N-Best generator, we observe that the presence of obstacles can indirectly result to the “trimming” of the search space and, so, reduce the execution time. In the E.A. generator, its execution time may depend on its probabilistic evolutionary process and the size of the search space, but it is relatively stable for similar scenarios. Lastly, the E.A. scales up better than the S.A.H.C. and N-Best algorithms, because, as the complexity of its task grows, the computing time required grows in a decreasing rate. If safety is not an immediate concern and the problem is complex, bounding the size of the E.A.'s search space can result in a path generation process much faster than the one of a S.A.H.C. or an N-Best algorithm.

In sum, for less complex scenarios the greedy S.A.H.C. and N-Best generators are recommended, because of their precision and efficiency. As the complexity of the scenarios grows, the probabilistic E.A. generator becomes increasingly preferable, because in large scales it is much more efficient than exhaustive searches.

5. CONCLUSIONS

My main contribution is proposing that the path optimization algorithm is not applied to the path's waypoints, but to the search space of the Bézier lines' control points, which are the parameters that characterize the lines' curvature. This search space exploits the inherent properties of Bézier curves, ensuring that all candidate paths are implementable by skid-steered rovers, and also avoid obstacles and make progress towards the goal. A further contribution is testing prominent algorithms from the optimization literature, to establish that evolutionary algorithms operate best in this search space, finding solutions in extremely challenging terrain and within reasonable computation time.

A future direction of my research can be to involve into the path selection process all three traversability parameters of a terrain that can be remotely sensed (slope, height, ruggedness), instead of only slope. Even if a terrain region has a relatively small inclination, it may be challenging because of its ruggedness and unevenness. Updating our path generators to also consider height and ruggedness would lead to the generation of the safest path with respect to the overall traversability of the terrain.

REFERENCES

[1] P. McGarey, D. Yoon, T. Tang, F. Pomerleau, and T. D. Barfoot, "Developing and deploying a tethered robot to map extremely steep terrain", Journal of Field Robotics, vol. 35, no. 8, Dec. 2018.

[2] C.-T. Chen, C.-C. Feng, and Y.-A. Hsieh, "Design and realization of a mobile wheelchair robot for all terrains", Journal of Advanced Robotics, vol. 17, no. 8, 2003.

[3] K. Inoue and M. Kaminogo, "Steep slope climbing using feet or shins for six-legged robots", in Proceedings of the 10th Asian Control Conference (ASCC 2015), 2015.

[4] J. Ge, A. Calderón, and N. Pérez-Arancibia, "An earthworm-inspired soft crawling robot controlled by friction", in Proceedings of the 2017 IEEE Intl Conference on Robotics and Biomimetics (ROBIO), 2017.

[5] D. Dunlap, W. Yu, E. G. Collins, and C. V. Caldwell, "Motion planning for steep hill climbing", in Proceedings of ICRA 2011.

[6] C. Ordonez, N. Gupta, O. Chuy, and E. G. Collins, "Momentum based traversal of mobility challenges for autonomous ground vehicles", in Proc. ICRA 2013.

[7] N. Gupta, C. Ordonez, and E. G. Collins, "Dynamically feasible, energy efficient motion planning for skid-steered vehicles", Autonomous Robots, vol. 41, 2017.

[8] M. Effati and K. Skonieczny, "Circular ARC-based optimal path planning for skid-steer rovers", in Proceedings of the 2018 IEEE Canadian Conf. on Electrical and Computer Engineering (CCECE). IEEE, Aug. 2018.

[9] Wolfram Mathword, "Bézier curve", http://mathworld.wolfram.com/BezierCurve.html, last accessed Feb 2019.

[10] M. Wermelinger, P. Fankhauser, et al., "Navigation Planning for Legged Robots in Challenging Terrain", in Proceedings of IROS 2016, October 2016.

ΔΙΠΛΩΜΑΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

Named Entity Recognition and Linking in Greek Legislation

Iosif E. Angelidis (iosang@di.uoa.gr)

ABSTRACT

We show how entity recognition in Greek legislation texts can be achieved by utilizing a named entity recognizer (NER). Our work is the first of its kind for the Greek language in such an extended form and one of the few that examines legal text. We apply grid search on multiple neural network architectures and combination of hyper-parameters to maximize the efficiency of our approach. We show that, utilizing a big legal corpus we built word/token-shape embeddings using Word2Vec, and finally achieve 86% accuracy on average in recognition of organizations, legal references, geographical landmarks, persons, geo-political entities (GPEs) and public documents. The evaluation of our methodology is based on the metrics of precision, recall, F_1-score per entity type for each neural network. Finally, we measure the ratio of correctly guessed links for the interlinking of RDF datasets produced by our approach with well-known public datasets and how new knowledge can be inferred indirectly by our approach from DBpedia, ELI (European Legislation Identifier) and GAG (Greek administrative geography) of Kallikratis.

Keywords: Named Entity Recognition and Linking, Legislative Knowledge Representation, Entity Reference Representation, Linked Open Data, Deep Learning, Entity Generation

Advisors

Manolis Koubarakis, Professor (National and Kapodistrian University of Athens) and Ilias Chalkidis, PhD Candidate (Athens University of Economics and Business)

1. INTRODUCTION

Recently, there has been an increased interest in the adaptation of Artificial Intelligence technologies to the legal domain including text processing, knowledge representation and reasoning. Legal text processing [4] is a growing research area, comprising of tasks such as legal question answering [20], legal entity extraction [12, 11] and legal text generation [2]. The same applies to the area of legal knowledge representation, where new standards have been developed and started to be adopted based on semantic web technologies. Relevant contributions here are the European Legislation Identifier (ELI) [14, 15, 16] for legislation, the European Case Law Identifier (ECLI) [25, 1] for case law, as well as Legal Knowledge Interchange Format (LKIF) [10, 18] and LegalRuleML [5, 6] for the codification of advanced legal concepts, such as rules and norms. The research community aims to develop tools and applications to help legal professionals (e.g., judges, lawyers, etc.) as well as ordinary citizens. Based on these principles our group created Nomothesi@[2] [13], a platform which makes Greek legislation available on the Web as linked data to aid its sophisticated querying using SPARQL and the development of relevant applications.

Deepening this effort in order to build a bridge, as a point of reference, between those relative research fields of data science (natural language processing and semantic web), we developed a Named Entity Recognizer (NER) and Linker (NEL) for Greek legislation. For the first task, we will compare and evaluate state-of-the-art neural architectures (RNNs) to recognise the following types of entities: persons, organizations, geopolitical entities, legal references, geographical landmarks and public document references from Greek legislation. We deploy our best entity recognizer on the Greek legislation dataset [13] and produce new entity knowledge encoded in RDF using a novel vocabulary. Given those triples, we use hand-crafted rules and the entity linking framework Silk [8, 7] in order to normalize and link the extracted textual references with entities in public open datasets (Greek administrative units and Greek politicians). We also publish a new RDF dataset for Greek geographical landmarks, which can usually be noted in legislation related to urban, rural and environmental planning. The main contributions are listed below:

- We study the task of named entity extraction in Greek Legislation by applying and evaluating state-of-the-art neural architectures [11], while we also examine a somewhat more complicated one, which outperforms the rest of them even by a short margin.

[1] http://legislation.di.uoa.gr

- We introduce a novel RDF vocabulary for the representation and linking of textual references to entities in Greek legislation. We consider RDF as a single data model for representing both metadata of a legislative document and knowledge that is encoded in the text.
- We deploy Nomothesi@ NER, based on the best model BILSTM-BILSTM-LR in the Greek legislation dataset and produce new data for entity references, that we describe using the new RDF vocabulary.
- We link the references with datasets using rule-based techniques and the Silk framework.
- We make publicly available a new benchmark dataset of 276 annotated legal documents, which can be reused to train and test different algorithms related to named entity recognition and linking. Pre-trained word embeddings specialized in Greek legal text are also provided.
- We generate a new dataset of Greek geographical landmarks based on the results of Nomothesi@ NER by applying heuristic rules. In a research project that our group has started this dataset will be enhanced further with additional geographical information (e.g., spatial relations and geometries) of the landmarks in order to support a service informing professionals, such as landscape engineers, as well as ordinary citizens about legislation that refers to specific geographical areas of Greece.
- Based on the above procedures, we augment the knowledge base and the querying capabilities of the Nomothesi@ platform in two significant ways: tracing legislation citation networks and searching using entity-based criteria.

This work is the first of its kind for the Greek language in such an extended form and one of the few that examines legal text in a full spectrum for both recognizing and linking entities. A publication [3] of this work was presented by the author of this thesis in JURIX 2018.

2. ANNOTATION AND DATASETS

When preparing datasets for NLP training, we need to provide examples of tokens and their labels so that we can feed this information into a neural network to properly train. To this end, the community has developed some tools dedicated to the task of *annotation*. For our purposes, we focus on *brat* (brat rapid annotation tool) [26]. Simply put, brat accepts as input a set of txt files, visualizing in a robust web client. Then, we can define classes of entities and any relations linking them as possible annotation labels. To prepare the datasets, all we need to do is annotate the tokens that we wish to give a label to; all this information is being written in ann files which contain lines with information such as the annotation id, the actual text, the class and the offsets (start and end).

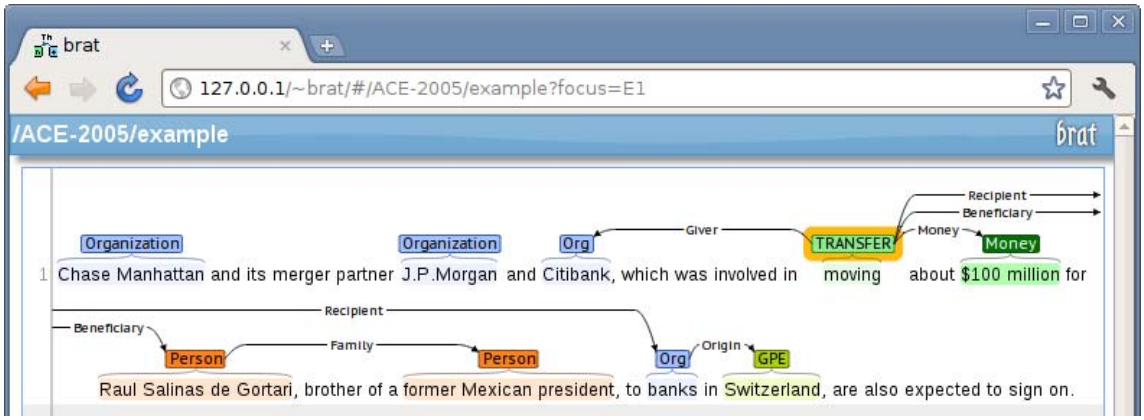


Figure 15: Visualization in brat.

The benchmark datasets for our experiments contain 276 daily issues for class A and D of the Greek Government Gazette over the period 2000-2017. Every issue contains multiple legal texts. Class A issues concern primary legislation published by the Greek government (e.g., laws, presidential decrees, ministerial decisions, regulations, etc.). Class D issues concern decisions related to urban, rural and environmental planning (e.g., reforestations, declassifications, expropriations, etc.).

We uniformly splitted the issues across training (184, 60%), validation (45, 20%), and test (47, 20%) in terms of publication year and class. Thus, the possibility of overfitting due to specific linguistic idiosyncrasies in the language of a government or due to specific entities and policies is minimal. We annotated all of the above documents for the 6 entity types that we examine, using *brat*.

3. NAMED ENTITY RECOGNITION

The main reason (BI)LSTMs (which are a more advanced form of RNN networks) are used for NLP is their ability to deal with information memorization and structure. Numerous examples such as Andrej Karpathy's[3] show many such applications. Examples involve teaching an RNN to learn English words and write Shakespeare parts on its own, syntactic structures from Wikipedia, writing Latex code that compiles or even writing Linux code.

Furthermore, the work of Chalkidis et al. [12, 11] has shown how BILSTM models can be applied on contracts to extract useful information. Adapting and evolving these techniques, we endeavour to achieve information and entity extraction from Greek legislation documents, expecting similar success in the process.

[2] See http://karpathy.github.io/2015/05/21/rnn-effectiveness/

3.1 Workflow

Let's begin by showcasing the summarized workflow of our approach, since that will make the following sections easier to comprehend:

1. We begin by taking a set of Greek legislation documents in PDF format, convert them into text and prepare the data so that each line contains a single sentence.
2. We tokenize the text so that each token is a single word. Punctuation are also tokens (with the exception of punctuation used in abbreviations).
3. We conduct `Word2Vec` and/or `FastText` training to obtain the word embeddings necessary to run neural network experiments.
4. We manually annotate greek legislation documents of the National Printing House using brat[4] [26] so that we can begin supervised training.
5. We feed the word embeddings in addition to embeddings shapes for each token into each of the four proposed models and do grid search so that we can determine the optimal set of parameters.
6. We evaluate the performance of the neural networks for all parameters calibrated during grid search.

3.2 Word Embeddings

Since we need to feed tokens into the neural network as well as for `Word2Vec/FastText` training, we need to tokenize the text obtained from the original PDF format. It is highly likely that we encounter certain punctuation like quotation marks, full stops, commas etc., all of which need to be artificially separated from any token/word they are next to. To achieve this, we have built a Tokenizer module that is based on NLTK[5]. However, we had to manually handle special cases like the above since the parsing tree provided by the library handles punctuation slightly differently. Furthermore, we convert all digits encountered in the text into "d", a necessary mapping for `Word2Vec` training. Lastly, it is necessary to normalize and capitalize the entire text and map all English words to a single word named "ENGLISH_WORD".

3.2.1 Word2Vec Training

In our work, we applied `Word2Vec` (skip-gram model) [23, 24] to an unlabelled corpus, which contains:

- 150,000 issues of Greek Government Gazette in the period of 1990-2017.

[3] http://brat.nlplab.org/

[4] http://www.nltk.org

• All pieces of legislation from the foundation of the Greek Nation in 1821 until 1990, which sum up to 50,000.

• 1,500 case laws published online by Greek Courts.

• Most EU Treaties, Regulations and Decisions, that have been translated in Greek and published in EUR-Lex.

• The Greek part of the European Parliament Proceedings Parallel Corpus.

We produced 100-dimensional word embeddings for a vocabulary of 428,963 words (types), based on 615 million tokens (words), included in the unlabelled corpus. We used Gensim's implementation of `Word2Vec` (http://radimrehurek.com/gensim/), with 10 minimum occurrences per word, 20 epochs and default values for other parameters. Out of vocabulary words were mapped to a single "UNK" embedding. The `Word2Vec` model training was carried out on a computer with an Intel®Xeon®E5-4603 v2, with a CPU frequency of 2.20GHz, a 10.24 MB L3 cache, a total of 128 GB DDR3 1600 MHz RAM and the Linux Debian 8.6 (Jessie) x86 64 OS.

3.2.2 FastText experimentation

We also experimented with publicly available generic pre-trained 200-dimensional word embeddings, which have been built with `FastText` [9] (https://fasttext.cc), based on a much larger corpus with Greek Wikipedia articles. As we will show, the experimental results were worse in specific entity types extracted by our neural networks, possibly because legal expressions are under-represented (or do not exist) in generic corpora (e.g., Wikipedia or news articles).

3.3 Token Shape Embeddings

We use token shape embeddings [11, 19] that represent the following seven possible shapes of tokens:

• token consisting of alphabetic upper-case characters, possibly including periods and hyphens (e.g., "ΠΡΟΕΔΡΟΣ", "Π.Δ.", "ΠΔ/ΤΟΣ")

• token consisting of alphabetic lower-case characters, possibly including periods and hyphens (e.g., "νόμος", "ν.", "υπερ-φόρτωση")

• token with at least two characters, consisting of an alphabetic upper-case first character, followed by alphabetic lower-case characters, possibly including periods and hyphens (e.g., "Δήμος", "Αναπλ.")

• Token consisting of digits, possibly including periods and commas (e.g., “2009”, “12,000”, “1.1”)

• Line break

• Any token containing only non-alphanumeric characters (e.g., “.”, “e”)

• Any other token (e.g., “1o”, “OIK/88/4522”, “EU”)

In general, the shape (form) of its token relies on the existence and relative position of alphabetic characters, digits and punctuation. Intuitively, this information is going to help the neural network conduct entity recognition more efficiently since we provide word embeddings and shapes for each token.

3.4 Hyper-parameter tuning

Based on experimentation, the pre-trained word embeddings are not updated during training on the labelled dataset, while in contrast token shape embeddings are not pre-trained.

The corresponding shape vectors are being learned during the actual training. We used Glorot initialization [17], binary cross-entropy loss, and the Adam optimizer [21] to train the recognizers with early stopping by examining the validation loss. Hyper-parameters were tuned by grid-searching the following sets, and selecting the values with the best validation loss: hidden units {100, 150}, batch size {16, 24, 32}, dropout rate {0.4, 0.5}.

A neural network, especially one with multiple layers, consists of millions of parameters and optimizing all of them is nearly impossible. We focus on the dropout percentage (dropout is the act of dropping a percentage of the network’s units and retrain them so that all neurons remain active and not biased) and batch size (number of samples propagated through the network, large value indicates faster training but less accuracy usually). The neural networks are trained for 30 epochs. The training was carried out on a computer with an Intel®Core™i5-7600, with a CPU frequency of 3.50GHz, 6.144 MB L3 cache, a total of 32 GB DDR4 2400 MHz RAM, an AORUS GeForce®GTX 1080 Ti with 11264 MB of memory, 3584 CUDA cores and the Linux Ubuntu Gnome 16.04.3 LTS (Xenial Xerus) x86 64 OS. The `Word2Vec` embeddings are vectors of 100 dimensions. Our neural network utilizes Python’s library of Keras 2.1.3[6], with tensorflow-gpu 1.4.1[7] as its backend.

[5] https://keras.io/

[6] https://www.tensorflow.org/

3.5 Evaluation

For each of the four methods we measured the performance on precision, recall, and F_1 scores measured per token. As suggested in [12], an evaluation per element, meaning per entity, can provide a more delicate estimation of each method's performance. Regardless, the complex syntax of the legislation text and more specifically groupings of multiple entities in long phrases (e.g., "The municipalities of Athens, Dafnis-Imittou and Varis-Voulas-Vouliagmenis will organize [...]") does not provide a clear segmentation between the individual entities[8] (e.g., Municipality of Athens, Municipality of Dafni-Imittos, Municipality of Varis-Voulas-Vouliagmenis), so that we may rely on for such a high-order evaluation. Table 1 lists the results of this group of experiments (the numbers are averaged over 5 runs of experiments).

Table 1: Precision (P), Recall (R), and F_1 score, *measured per token*. Best F_1 per entity type shown in bold font.

Entity Type	BILSTM-LR			BILSTM-LSTM-LR			BILSTM-CRF			BILSTM-BILSTM-LR		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Person	0.89	0.90	0.89	0.89	0.94	**0.91**	0.80	0.92	0.90	0.89	0.93	**0.91**
Organization	0.77	0.73	0.75	0.77	0.78	0.77	0.72	0.74	0.73	0.78	0.77	**0.78**
GPE	0.80	0.87	0.84	0.83	0.89	0.86	0.81	0.86	0.83	0.84	0.90	**0.87**
GeoLandmark	0.67	0.82	0.73	0.72	0.86	**0.78**	0.64	0.83	0.72	0.70	0.86	0.77
Legislation Ref.	0.85	0.81	0.83	0.87	0.85	**0.86**	0.80	0.79	0.80	0.88	0.85	**0.86**
Public Document	0.81	0.75	0.78	0.85	0.81	0.82	0.72	0.75	0.74	0.84	0.81	**0.83**
Macro AVG	0.82	0.84	0.83	0.84	0.87	**0.86**	0.79	0.84	0.81	0.85	0.87	**0.86**

The results are highly competitive for all the examined methods. The best results based on the macro-averaged F_1 are coming from both BILSTM-LSTM-LR and BILSTM-BILSTM-LR (0.86), which indicates that the extra LSTM chains, which deepen the model, expand its capacity even by a short margin, compared to BILSTM-LR (0.83) and BILSTM-CRF (0.81). The deficiency of the current NER state-of-the-art method BILSTM-CRF, which has been validated across all possible hyper-parameter sets, is quite impressive. We strongly believe that this issue is strongly correlated with the complicated references of geographical landmarks, legislation references and public documents references, especially in cases with entity reference groupings under a single keyword as demonstrated above.

[7] This shortcoming is true for both IO and BIO annotation schemes, which have been widely applied in sequence labelling tasks.

Table 2: Precision, Recall, and F_1 score for FastText, *measured per token* with BILSTM-BILSTM-LR.

Entity Type	Precision	Recall	F_1-score
Person	0.89	0.88	0.88
Organization	0.75	0.7	0.72
GPE	0.85	0.78	0.81
GeoLandmark	0.64	0.76	0.7
Legislation Ref.	0.82	0.82	0.82
Public Document	0.77	0.74	0.76
Macro AVG	0.81	0.81	0.81

Further on, we are going to rely on the BILSTM-BILSTM-LR recognizer based on the fact that it outperforms the BILSTM-LSTM-LR by 1% in F 1 in Organizations (0.78 vs 0.77), Geopolitical Entities (0.87 vs 0.86) and Public Documents (0.83 vs 0.82), while it is only 1% worse in Geographical Landmarks (0.77 vs 0.78). Considering the generic `FastText` pre-trained embeddings instead of our domain-specific ones, leads to a macro-averaged F_1 of 0.81 for the best reported method BILSTM-BILSTM-LR (Table 1), especially in the latter four categories, in which domain knowledge matters the most (e.g., geographical aspects and codification of documents).

4. NAMED ENTITY LINKING

Let's begin by showcasing the summarized workflow of our approach, since that will make the following sections easier to comprehend (each step will be analyzed further):

1. We apply post-processing techniques with hand-written rules and regexes to normalize and process the extracted entities into presentable labels.

2. Alongside the labels, we generate RDF data regarding the named entities. Useful properties include the passage in which they were found, their position in the text (for a web-page annotation).

3. We interlink *geo-political entities* (GPEs), persons and legislation references with Kallikratis (GAG), Dbpedia persons and ELI, respectively with the Silk framework. An intermediate dataset consisting of `owl:sameAs` is generated, as a result.

4. We manually generate a dataset of landmarks which are usually noted in legislation related to urban, rural and environmental planning and, based on heuristic rules and relative position within passages, interlink them with belongs_to relations to corresponding GPEs.

4.1 Textual entity references vocabulary

The first step towards linking entity references extracted (by the Named Entity Recognizer) with the entities described in public open datasets is to represent those references using the RDF specification. The *legal text* of a document contains subdivisions (essentially, passages of individual laws), which we define as *LegalResourceSubdivisions* based on the Greek legislation ontology. Since some of those contain text, it is also possible to contain (*has_reference to*) a *Reference* to an entity (e.g., a law passage referring to a specific law that it modifies). This *reference* is realized in an interval of characters. In other words, it begins and ends on specific sequential characters inside the text of the subdivision. This Reference most likely refers to (or in another sense is *relevant_for*) an Entity, which is probably described in open public datasets. Therefore, a *LegalResourceSubdivision* contains references to persons, administrative units and legal resources (e.g., laws, decisions etc.).

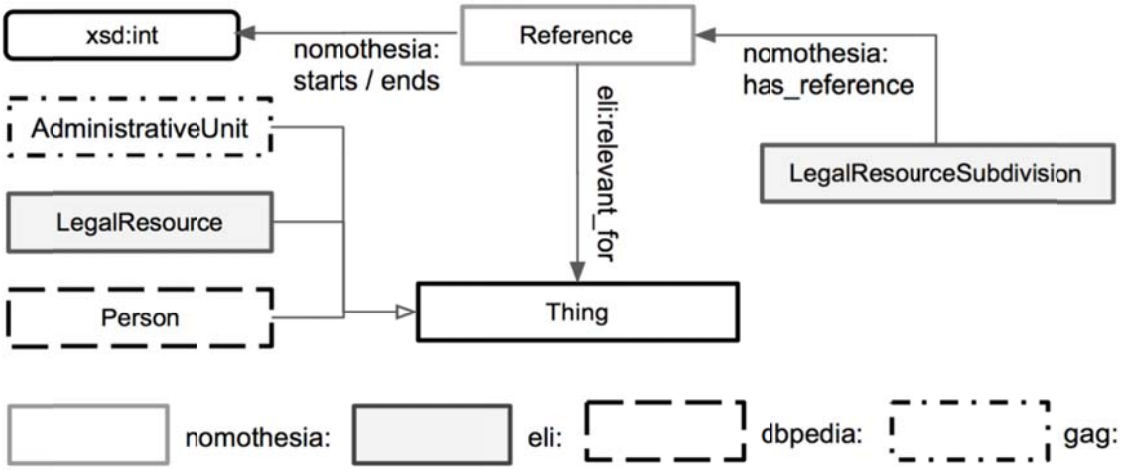


Figure 2: Textual reference RDF vocabulary.

4.2 Using Silk and heuristics to generate owl:sameAs links

We linked legal references with legal documents provided by the Greek legislation dataset[9]. We based on heuristic rules to directly interpret the relevant URI by capturing the type, year of publication and the serial number.

We linked person references with Greek politicians retrieved from the Greek DBpedia [10] dataset and geopolitical entity references with the Greek administrative units as they are described in the Greek Administrative Geography (GAG) dataset. For both entity types, we proceed in interlinking the corresponding datasets using the Silk framework. We experimented with two different textual linking operators: Levenshtein and Substring distance [27] over

[8] Published in http://legislation.di.uoa.gr/data.

[9] http://el.dbpedia.org/

the `rdfs:label` values provided by each dataset. For the case of the Greek Administrative Units, we also provided the type of the administrative units (e.g., local community, municipality, region, etc.) based on the naming conventions that we identified in the validation part of the labelled dataset.

For each interlinking method that we tried, we examine the performance of the interlinking in terms of *precision*, *recall*, and F_1 score *measured per entity pair* on the test part of our labelled dataset. Here, *true positives* (TP) are references correctly paired with an entity of each set, *false positives* (FP) are references incorrectly paired with entities, and *false negatives* (FN) are references incorrectly not paired with the relative entities of the examined sets. The acceptance threshold for both linking operators was tuned on the validation part of our datasets, while the entity pairs provided are those presented in the test part. Table 3 lists the results for this group of experiments.

Figure 3: The interlinking process in Silk for persons and GPEs.

4.3 Evaluation

Linking persons was a great challenge for our system, mainly because legislators and the publication office tend to refer to a person's first name by its initials (e.g., A. Tsipras), thus a fair amount of person references have been misclassified (precision: 0.71) for persons with the same surname. We successfully linked the geopolitical entities with the Greek administrative units (F_1: 0.92). Minor issues are related to the segmentation of compound references of multiple administrative units. The results for legislation references are excellent (F_1: 0.98), while a short margin of documents is mis-linked due to the fact that ministerial decisions do not have a standard codification (neither standard reference pattern), which vary from one ministry to another.

Table 3: Precision (P), Recall (R), and F_1 score, *measured per entity pair*.

Metrics	Linking technique								
Entity Type	Rules			Levenshtein			Substring		
	P	R	F_1	P	R	F_1	P	R	F_1
Person	-	-	-	0.99	0.55	0.71	0.9	0.68	**0.77**
GPE	-	-	-	0.99	0.79	0.88	0.95	0.92	**0.94**
Legislation Ref	0.99	0.97	**0.98**	-	-	-	-	-	-

Greek geographical landmarks are a major asset for our legal recognizer since they are related to planning and architectural interests. However, there is no such public dataset to interlink between the references and the actual entities. Therefore, we produce a new dataset by applying lingual heuristics in order to create a set of unique geographical landmarks, resulting in the ontology shown below:

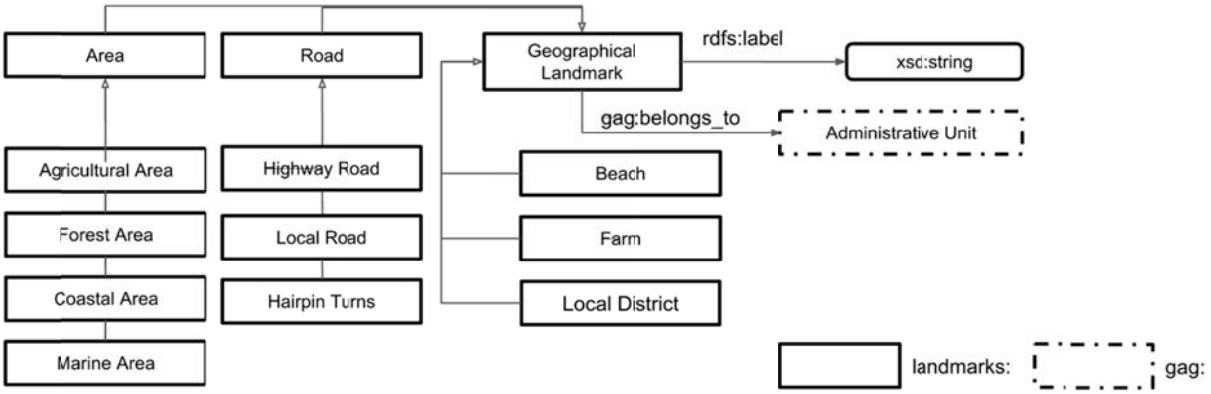


Figure 4: Geographical Landmark RDF vocabulary.

Further on, we interlink the new dataset with the Greek administrative units in case there is a connection between them (belongs_to) indicated in terms of text (e.g., “Beach Kavouri at Municipality of Varis-Voulas-Vouliagmenis”).

5. DEMONSTRATING THE NER/NEL’S FUNCTIONALITY

5.1 Legislation citation networks

A legal professional may retrieve citation networks built around a legal document, which most likely include legal documents in the same context (see Figure 5).

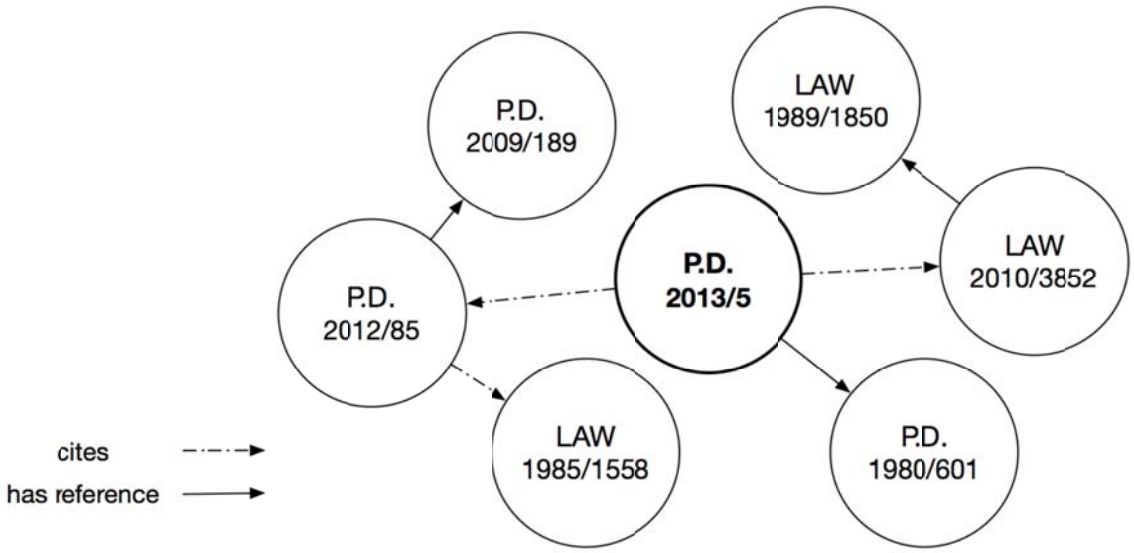


Figure 5: Citation network based on citations and references around Presidential Decree 2013/5.

5.2 Entity-based search

Based on the above, we have the ability to pose queries against the resulting RDF graph. We provide a few sample queries, given in natural language together with their expression in SPARQL in the portal's website.

6. CONCLUSION AND FUTURE WORK

Overall, we developed, tested and evaluated a Named Entity Recognition and a Named Entity Linking component, applied on Greek legislation. Greek is a challenging language for NLP tasks, while the additional noise from external sources (since the original corpus of documents is only available in PDF format) provided an interesting challenge to tackle.

Regarding the NER component, we evaluated all of the above LSTM-based methods in the task of Named Entity Recognition in a Greek legislation dataset, which we made publicly available for further academic research. The process was challenging and lengthy, as we had to convert PDF files into TXT format, process them so that they are suitable for training, manually annotate a subset of the documents to generate the test, train and validation parts of the datasets, before being able to conduct our experiments. As reported above, our experiments yielded some interesting and even unexpected findings.

Regarding the NEL component, we evaluated entity-linking between textual references and entities from open third-party datasets. Obtaining links is important as we can complement the information of the entities extracted from the text with their corresponding/dedicated matches on popular datasets.

Finally, we introduced and applied a novel vocabulary for the representation of textual references and we generated a new dataset for Greek geographical landmarks. As explained before, rural/architectural information of this kind has never been extracted into a dataset of any kind; therefore it is a significant contribution as it provides us with numerous capabilities.

Our future plans include further experimentation on the LSTM-based methods using word embeddings trained with the FastText algorithm, which considers sub-words information. We consider that it would be beneficial based on the fact that the Greek language includes multiple declensions in the indication of numbers, cases (nominative, subjective, genitive, possessive), and genders. For the same reasons, we are also planning to replace the shape embeddings with a dynamic character-level RNN or CNN model, to embed information relevant to token shapes, prefixes, suffixes, as described by Ma and Hovy [22].

A character-level RNN or CNN model will also be examined as an alternative method (operation) for entity linking.

Another interesting potential direction is the introduction of a more complicated annotation format with richer sets of labels, based on the principles of BIO tags, in order to address the complexity of the legal text. We also endeavour to extract (recognize) more geospatial information such as coordinates, presented in tables, or extracting relations between landmarks to augment the information in the newly generated dataset.

REFERENCES

[1] T. Agnoloni, L. Bacci, G. Peruginelli, M. van Opijnen, J. van den Oever, M. Palmirani, L. Cervone, O. Bujor, A. A. Lecuona, A. B. García, L. D. Caro, and G. Siragusa, "Linking european case law: Bo-ecli parser, an open framework for the automatic extraction of legal links," in *JURIX*, 2017.

[2] W. Alschnerd and D. Skougarevskiy, "Towards an automated production of legal texts using recurrent neural networks," in *Proceeding of the 16th International Conference on Artificial Intelligence and Law*, (London, UK), pp. 159–168, 2017.

[3] I. Angelidis, I. Chalkidis and M. Koubarakis, "Named Entity Recognition, Linking and Generation for Greek Legislation," in *JURIX* 2018.

[4] K. Ashley, *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age*. Cambridge University Press, 2017.

[5] T. Athan, H. Boley, G. Governatori, M. Palmirani, A. Paschke, and A. Wyner, "OASIS LegalRuleML," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law*, (Rome), pp. 3–12, 2013.

[6] T. Athan, G. Governatori, M. Palmirani, A. Paschke, and A. Z. Wyner, "LegalRuleML: Design principles and foundations," in *Reasoning Web. Web Logic Rules*, 2015.

[7] C. Bizer, A. Jentzsch, and R. Isele, "Silk - generating rdf links while publishing or consuming linked data," in *International Semantic Web Conference*, (Shanghai, China), 2010.

[8] C. Bizer, A. Jentzsch, and R. Isele, "Silk server - adding missing links while consuming linked data," in *1st International Workshop on Consuming Linked Data*, (Shanghai, China), 2010.

[9] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *CoRR*, vol. abs/1607.04606, 2016.

[10] J. Breuker, R. Hoekstra, A. Boer, K. van den Berg, R. Rubino, G. Sartor, M. Palmirani, A. Wyner, and T. Bench-Capon, "OWL ontology of basic legal concepts (LKIF-core), deliverable 1.4," tech. rep., ESTRELLA, 2007.

[11] I. Chalkidis and I. Androutsopoulos, "A deep learning approach to contract element extraction," in *Proceedings of the 30th International Conference on Legal Knowledge and Information Systems*, (Luxembourg), pp. 155–164, 2017.

[12] I. Chalkidis, I. Androutsopoulos, and A. Michos, "Extracting contract elements," in *Proceedings of the 16th International Conference on Artificial Intelligence and Law*, (London, UK), pp. 19–28, 2017.

[13] I. Chalkidis, C. Nikolaou, P. Soursos, and M. Koubarakis, "Modeling and querying greek legislation using semantic web technologies," in *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, pp. 591-606, 2017.

[14] J. Dann, "European Legislation Identifier "ELI"," tech. rep., European Commission, 2014.

[15] ELI Task Force, *ELI - A technical implementation guide*, 2015.

[16] ELI Task Force, *ELI implementation methodology: Good practices and guidelines*, 2015.

[17] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010* (Y. W. Teh and D. M. Titterington, eds.), vol. 9 of *JMLR Proceedings*, pp. 249–256, JMLR.org, 2010.

[18] R. Hoekstra, J. Breuker, M. Di Bello, and A. Boer, “LKIF core: Principled ontology development for the legal domain,” in *Proceedings of the 2009 Conference on Law, Ontologies and the Semantic Web: Channelling the Legal Information Flood*, (Amsterdam, The Netherlands, The Netherlands), pp. 21–52, IOS Press, 2009.

[19] D. Jurafsky, *Speech and language processing: An introduction to natural language processing*. Prentice Hall, 2018.

[20] M. Kim and R. Goebel, “Two-step cascaded textual entailment for legal bar exam question answering,” in *Proceedings of the 4th Competition on Legal Information Extraction/Entailment*, (London, UK), 2017.

[21] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proceedings of the International Conference on Learning Representations*, (San Diego, CA), 2015.

[22] X. Ma and E. Hovy, “End-to-end sequence labeling via bi-directional LSTM-cNNs-CRF,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, (Berlin, Germany), pp. 1064-1074, 2016.

[23] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *CoRR*, vol. abs/1310.4546, 2013.

[24] T. Mikolov, W. Yih, and G. Zweig, “Linguistic regularities in continuous space word representations,” in *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA* (L. Vanderwende, H. D. III, and K. Kirchhoff, eds.), pp. 746-751, The Association for Computational Linguistics, 2013.

[25] M. V. Opijnen, “European Case Law Identifier: Indispensable Asset for Legal Information Retrieval,” in From Information to Knowledge (M. A. Biasiotti and S. Faro, eds.), vol. 236 of *Frontiers in Artificial Intelligence and Applications*, pp. 91–103, IOS Press, 2011.

[26] P. Stenetorp, S. Pyysalo, G. Topic, T. Ohta, S. Ananiadou, and J. Tsujii, “brat: a web-based tool for nlp-assisted text annotation,” in *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, April 23-27, 2012* (W. Daelemans, M. Lapata, and L. Màrquez, eds.), pp. 102–107, The Association for Computer Linguistics, 2012.

[27] G. Stoilos, G. Stamou, and S. Kollias, “A string metric for ontology alignment,” in *4th International Semantic Web Conference*, (Galway, Ireland), pp. 624–637, 2005.

Simulation of Nanoscale Roughness Evolution of Silicon Surfaces under Chlorine Plasmas

Antoniou Iro Maria (iantonio@di.uoa.gr, iro9ntoniou@gmail.com)

ABSTRACT

A surface model for Si etching by Cl_2 plasma is developed and coupled with a Monte Carlo (MC) modeling framework to predict the etching rate and the nanoscale roughness of Si surfaces. A critical parameter for the accuracy of the calculations is the ion-enhanced etching yield (number of atoms of the substrate removed per incident ion). The application of an etching yield expression from the literature to the MC framework leads to results which strongly deviate from the etching rate measurements. For a MC framework, a “nanoscopic” etching yield suitable to reproduce the “macroscopic” etching yield and rate (from literature) is required. In this work, the “nanoscopic” etching yield of the dominant etching mechanism is extracted by fitting the etching rate to available measurements. The MC framework reproduces well the experimentally measured dependence of the etching rate on the ion energy. Furthermore, the behavior of rms roughness versus ion energy is captured when the redeposition of the etching products is intense.

Keywords: Nanoscale Surface Roughness, Silicon, Modeling, Simulation, Ion-enhanced Etching, Monte Carlo

Advisor

George Kokkoris, Research Associate, Institute of Nanoscience and Nanotechnology, NCSR ‘Demokritos’

1. INTRODUCTION

1.1 Informatics and plasma etching

Microelectronics enables what informatics promises. Informatics, on the other hand, adds value even to highly developed chips. “It is the colossal flexibility afforded by the programmability of programmable chips that is the key to technological progress. And this can blossom only in the interplay between informatics and microelectronics” [1]. The fabrication flowchart of a chip in micro- and nanoelectronics involve a series of unit processes such as lithography (optical or e-beam), plasma etching, chemical or physical vapor deposition of thin films, ion implantation, and oxidation [2]. For the fabrication of a chip, one-third of the tens to hundreds of fabrication steps are typically plasma based [3]. A plasma is a collection of free charged and neutral particles that is, on the average, electrically neutral [3]. The technological plasmas being used in microelectronics are produced by the application of electrical and magnetic fields. The reactive neutral and charged particles produced in the plasma are indispensable for the etching of substrates, such as Silicon (Si) or polymeric substrates; Si is the most common substrate in micro- and nanoelectronics.

1.2 The importance of Surface Roughness in Microelectronics and Other Fields

Surface roughness is a usual outcome of the micro- and nanofabrication processes (see Section 1.3) and can be critical for several applications and pertinent fields. Firstly, it is important in micro- and nanoelectronics. Roughness on atomic- or on nanoscale deeply affects device (e.g. transistor) operation as it is of the same scale as the critical dimension (CD) of the etched features [4]. It is also important in microelectromechanical systems (MEMS). Even though the scale is larger, roughness has an impact on the fracture strength of microstructures and the friction between moving parts. Roughness also affects the wetting properties of the surfaces [5]. Nanoroughness can enhance antireflectivity not only for Si, but also for polymeric surfaces [6]. Roughness can also be critical for the interaction of surfaces with biological samples [7]. With a level of periodicity, the surface roughness, can be useful to magnetic storage [8], catalysis [9], and nanopatterning [10].

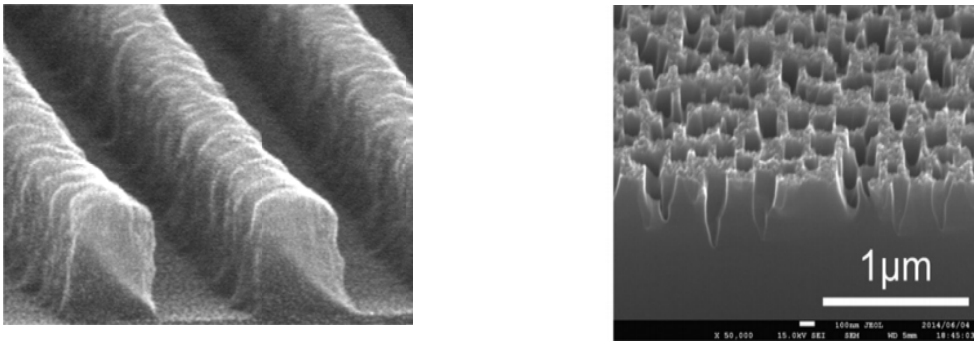
1.3 Plasma Etching Mechanisms

The origin of surface roughness during plasma etching of Si or other substrates is attributed to competitive “forces” between roughening and smoothing and comes from the non-uniformity of the etching rate on the surface of the substrate. Thus, it is useful to identify the etching mechanisms in a plasma environment. First of all, etching techniques are separated into wet and dry or

plasma etching techniques. Wet etching is more a chemical than a physical process. It is therefore, quite a selective and isotropic process. Plasma etching allows for both anisotropic and isotropic etching. Plasma etching can be categorized to chemical etching, physical sputtering, and ion-enhanced etching. Chemical etching is attributed to highly reactive neutral species chemically sensitive with specific substrates and leads to isotropic etching. Physical sputtering is the ejection of atoms or molecules from the surface due to energetic ion bombardment [3] and is not a selective process since the sputtering yield (number of substrate atoms removed per incident ion) depends only on the ion energy, the surface binding energy, and the masses of the targets and projectiles. Sputtering is also an anisotropic process strongly sensitive to the angle of the incidence of the ion. Last but not least, ion-enhanced (chemical) etching occurs when a target substrate is exposed to both reactive neutrals and ion bombardment. Among plasma etching mechanisms, ion-enhanced etching is the most promising for micro- and nanofabrication and essentially interesting for this thesis also. Ion-enhanced etching was introduced by Coburn and Winters [11]. They observed that the etching rate in the case of Si containing substrate (Si, SiN_3, SiO_2, SiC) etching under beams of XeF_2 and Ar^+ is higher than the etching rate by an XeF_2 or Ar^+ beam alone. In ion-enhanced etching, the ion bombardment boosts one of the steps of the chemical etching process. Alternatively, the ion bombardment may dislodge or sputter away etch byproducts which would otherwise tend to stay on a surface and impede the etching process [2]. Ion-enhanced etching is an anisotropic etching process usually with a much greater rate compared to physical sputtering. Selectivity is also better compared to physical sputtering. The etching yield for ion-enhanced etching depends on the ion energy as the etching yield for physical sputtering; it is also angle dependent. Besides the three mechanisms of material removal, simultaneous deposition may occur during plasma etching [12, 13].

1.4 Roughness of Silicon Surfaces: Evidence, Mechanisms, and Control

Surface roughness may be formed either on the sidewall of the etched feature (e.g. trench or hole), or on open areas samples. In the first case we refer to sidewall roughness which contributes to line edge roughness (LER) and line width roughness (LWR).



a) b)

Figure 1: a) Sidewall roughness – roughness on the sidewalls of trenches of polymeric lines [Scanning electron microscope (SEM) image]. b) Open area surface roughness of a Si surface (SEM image) [14]

The focus of this work is on the second case and on the formation of surface roughness during Chlorine plasma etching of open area Si substrates. Si is the substrate material of integrated circuits. There are a few previous works investigating the factors affecting the roughness formation. In particular, Petri et al. came up with an empirical relation for root mean square (rms) roughness [15]. At low ion flux the surface is smoothened by the neutral species, while at higher ion flux, the neutral species roughen the surface as ion induced effects are triggered [15]. Hwang et al. [16] studied the evolution of surface roughening of Si during etching by Chlorine (Cl_2) plasma. The outcome of the study was that the surface roughness depended on self-bias voltage and pressure. In addition, surface roughness increased as a function of the etching time. Sung and Pang [17] reported that both the etching rate and the surface roughness of Si samples increased with the increase of the rf power on the electrode. Kokkoris et al. [13] and Boulousis et al. [18] investigated the effect of the etching time and wafer temperature on the surface roughness of Si during etching with SF_6 plasma in a helicon type plasma reactor. Dual nanoscale morphology, as well as, almost linear increase of both rms roughness and correlation length versus etching time was observed in the experiment. A mechanism based on the deposition of etch-inhibitors was proposed for the explanation of the experimental behavior. Yin & Sawin [19] measured the evolution of roughness of Si surfaces in Ar plasma by atomic force microscopy (AFM) as a function of ion bombardment energy, ion impingement angle, and etching time in an inductively coupled plasma beam chamber. Their study demonstrated the importance of the angle of ion incidence on the surface roughness. The roughness became greater at grazing angles. Martin & Cunge [20] analyzed the roughness generated on c-Si (100) surfaces when etched in an inductively coupled industrial plasma source over a wide range of conditions and chemistries. They reported that plasma etching did not induce roughness on the Si surface; on the contrary, it smoothened the Si surface. The smoothing

was attributed to the fact that the hills of a rough surface received a higher flux of etchant radicals than the valleys. Nakazaki et al. [14] reported two surface evolution modes during Si etching in inductively coupled Cl_2 plasmas, i.e. the roughening and the smoothing mode. These modes depend upon the ion energy. The transition from the roughening to the smoothing mode is attributed to a change in the gas phase species (induced by the etching products) as the ion energy increases.

1.5 Literature Review of the Methods Used for the Evolution of the Surface Morphology

The methods for the evolution of surface morphology entail a discrete or a continuum description of the surface morphology (or profile) and a Monte Carlo (MC) model or a deterministic method for the surface (or profile) evolution. The string method [21], the method of characteristics [22], and the level-set method [21] are deterministic algorithms for the evolution of the surface morphology; in all these algorithms, the description of the surface morphology is continuum. The cell-based method [12, 24-25] is a stochastic algorithm based on a MC model and the description of the surface morphology is discrete. In the cell-based method, the surface morphology consists of cubic cells each of which may contain more than one atom or molecule (coarse graining). Cells can be removed or added depending on etching or deposition probabilities as defined by the etching yields and the sticking probabilities. In this work, we utilized a cell-based framework [12] to study the evolution of surface morphology of Si substrates in Cl_2 plasmas. A review of the cell-based frameworks is following. Sawin and coworkers developed a 3d modeling framework for polysilicon etching in inductively coupled Cl_2 and HBr plasmas [26]. They developed a “mixing-layer kinetics model” [26] and investigated the onset of surface roughening. Kushner and coworkers [25, 27-29] developed a 3d profile simulator for feature scale profile evolution and applied it in several cases of feature etching. Wang and coworkers [30] developed a (2+1)d MC simulation framework to study the scaling behavior of plasma etching of Si substrates. Ono, Eriguchi, and coworkers [24, 31-32] started from (1+1)d and finally developed a 3d profile evolution simulator and applied it in Si etching in chlorine-based plasmas. Surface roughness was calculated as a function of ion energy (or rf bias) and other parameters. Agreement with measured values was observed for ion energies up to 250 eV. Kokkoris et al. [12, 13] developed a (2+1)d MC simulation framework and studied the effect of ‘simultaneous-to-etching deposition’ (SIMED) on the surface morphology evolution. The presence of etch-inhibitors was found to contribute to the formation of periodic dots on the etched surface [12].

1.6 The Purpose of the Thesis

The purpose of the thesis is the development of a surface model for Si etching under Cl_2 plasma. The surface model will be incorporated in an available MC modeling framework [12] in order to calculate the evolution of nanoscale roughness of Si surfaces. The MC framework will be extended to properly treat the ion enhanced etching mechanism, which is dominant in etching of Si with Cl_2 plasma. The coupling of the surface model with the MC framework will be evaluated by a comparison to measurements [14] of the etching rate and roughness versus the ion energy. A MC framework coupled with a validated surface model is a very useful tool for the prediction of nanoscale roughness of plasma etched surfaces. It can shed light on the mechanisms of roughness formation and evolution and contribute to the design of recipes delivering the desired (per application) surface roughness.

2. MATHEMATICAL MODEL

2.1 The Surface Model

The main mechanisms of etching of Si in a Cl_2 plasma are a) physical sputtering, b) ion-enhanced etching, and c) chemical etching. Besides the critical species from the gas phase produced in the reactor bulk of a Cl_2 plasma [31], namely energetic chlorine ions (Cl^+) and atomic chlorine (Cl), the species joining the surface etching model are adsorbed chlorine species ($SiCl_x$(s), x=1,2,3,4), saturated ($SiCl_4$) and unsaturated ($SiCl_x$, x=1,2,3) etching products. The surface processes during etching of Si surfaces with Cl_2 plasmas are described in Table 1.

	Reaction	Process		Coefficient
1	Adsorption of neutral species	$SiCl_x(s)+Cl(g)\rightarrow SiCl_{x+1}(s)$	x = 0,1,2,3	S_x
2	Chemical etching	$SiCl_3(s)+Cl(g) \rightarrow SiCl_4(g)$ $SiCl_4(s) \rightarrow SiCl_4(g)$		α_{chem}
3	Ion enhanced etching	$SiCl_x(s) + Cl^+ \rightarrow SiCl_x(g)$	x=1,2,3,4	EY_{SiCl_x}
4	Physical sputtering	$Si(s) + Cl^+ \rightarrow Si(g)$		EY_{Si}
5	Redeposition	$SiCl_x(g) \rightarrow SiCl_x(s)$	x=0,1,2,3,4	S_d

Table 1: Surface processes during etching of a Si surface with a Cl_2 plasma [30]

Besides chlorination through Cl sticking on the surface (reaction 1 of Table 1), Cl can induce chemical etching on Si surfaces (reaction 2 of Table 1; defined by coefficient a_{chem}). a_{chem} depends on the number density of Cl atoms, the surface temperature, the doping concentration of Si, and the activation energy [33]. Cl^+ ions can induce ion-enhanced etching (reaction 3 of Table 1) and physical sputtering (reaction 4 of Table 1) of the Si surface. The etching yield for ion-enhanced etching depends on the ion energy, the angle of ion incidence, and the chlorination level, and reads [14, 31, 34]

$$EY_{SiCl_x} = \frac{x}{4} A_{SiCl_x} \left(\sqrt{E_+} - \sqrt{E_{th,SiCl_x}} \right) f_{SiCl_x}(\theta), \qquad (1)$$

where $E_{th,SiCl_x}$ is the threshold energy for ion-enhanced etching (12 eV) [36] and θ is the incident angle of ions (rads). The function f_{SiCl_x} expresses the effect of the angle of ion incidence on the etching yield and comes from Refs. [26, 31, 34]. The etching yield of physical sputtering depends on the ion energy and the angle of ion incidence as well [26, 31, 34].

Finally, the etch products can be redeposited on the Si surface (reaction 5 in Table 1) with a sticking probability, S_d. The etching products are assumed to be desorbed from the surface thermally following the cosine law [14, 31].

2.2 Monte Carlo Framework for the Evolution of Surface Morphology

A modeling framework for surface morphology evolution in (2+1)d [12] is exploited to apply the roughening mechanisms of Si surface under Cl_2 plasma. In the context of this framework, the etched film is represented by a lattice of cubic cells (see figure 2) and the solid on solid approximation [37] is considered. Particles with user defined angular distributions impinge on the cellular morphology. No particle-particle collisions are considered due to the high Knudsen number of the flow in the valleys of the surface morphology; the mean free path is large due to the low pressure conditions during plasma etching compared to the dimensions of valleys. The trajectory of each particle is calculated until sticking on a cell. The interaction of particles with the cells is defined by a) the sticking probability and b) the etching yield. MC method is used to sample stochastic variables from probability distributions defining the initial position and direction of the particle, the etching yield, the reemission probability, and the direction of reemitted particles. Statistical parameters of the surface morphology are extracted by a module for the characterization of the surface morphology.

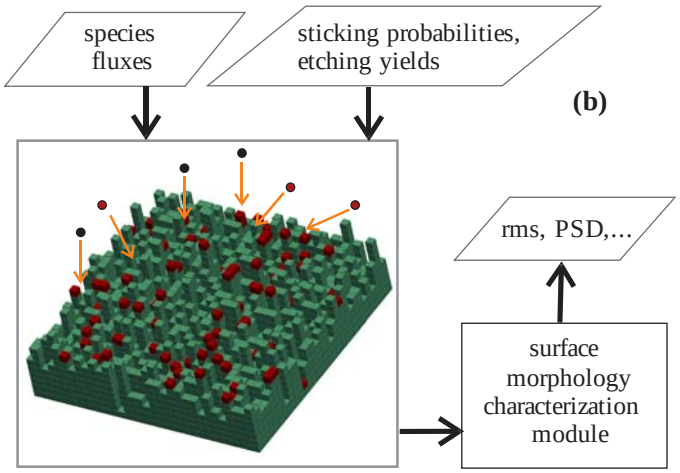


Figure 2: Schematic with the inputs and the outputs of the MC modeling framework

3. EXPERIMENTS & RESULTS

3.1 Case Study

All values used for the inputs of the models are typical for inductively coupled plasma (ICP) and electron cyclotron resonance (ECR) discharges [32]. The conditions (values of inputs) are [32]: Ion energy, E_+ = 100 eV, standard deviation of angular distribution of ions, σ = 0.05 rad, gas temperature, T_g =500 K. The etched surface is plane n-type Si(100) surface with atomic density equal to 5×10^{22} cm^{-3}. The dopant concentration is $N_e = 1.01\times10^{18}$ cm^3 and the surface temperature is T_s = 320 K. The ion (Cl^+) flux, Γ_+, is 10^{20} $m^{-2}s^{-1}$ and the ratio of neutral (Cl) to ion (Cl^+) flux is 100. The angle of the main direction of the ions arriving on the surface with the normal to the surface is 0^o. Table 2 summarizes the values of the parameters used for the runs.

case	A_{SiCl_x} ($eV^{-0.5}$)	A_{Si} ($eV^{-0.5}$)	a_{chem}	x_+	S_d	E_+ (eV)	f_{SiCl_x}	Surface dimension (nm^2)	Cell dimension (nm^3)	Number of runs
0001	1.3061	0	0	1	1	100	Eq. 2	128×128	0.271^3	6
0002	0.353	0.0356	0.0009	0.0099	0.05	100	Eq. 2	50×50	0.271^3	3
0003	0.353	0.0356	0.0009	0.0099	0	100	1	50×50	0.271^3	3
0004	0.3354 – 0.7237	0	0	1	0 – 1	100	1	64×64	1	1 per S_d value
0005	1.3061 – 2.0121	0	0	1	0 – 1	100	Eq. 2	64×64	1	1 per S_d value
0006	1.3061 – 1.9768	0	0.0009	0.0099	0 – 1	100	Eq. 2	64×64	1	1 per S_d value
0007	1.3061 – 1.9768	0.0178 – 0.0338	0.0009	0.0099	0 – 1	100	Eq. 2	64×64	1	1 per S_d value
0008	0	0.0178 – 0.0338	0	1	0 – 1	100	-	64×64	1	1 per S_d value
0009	1.9768	0	0	1	0.05	50	Eq.	128×	0.271^3	5

							2	128		
0010	1.9768	0	0	1	0.05	100	Eq. 2	128× 128	0.271^3	5
0011	1.9768	0	0	1	0.05	150	Eq. 2	128× 128	0.271^3	5
0012	1.9768	0	0	1	0.05	200	Eq. 2	128× 128	0.271^3	5
0013	1.9768	0	0	1	0.05	250	Eq. 2	128× 128	0.271^3	5
0014	1.9768	0	0	1	0.05	300	Eq. 2	128× 128	0.271^3	5
0015	1.9768	0	0	1	0.05	400	Eq. 2	128× 128	0.271^3	5
0016	1.9768	0	0	1	0.05	500	Eq. 2	128× 128	0.271^3	5
0017	1.9768	0	0	1	1	50	Eq. 2	128× 128	0.271^3	5
0018	1.9768	0	0	1	1	100	Eq. 2	128× 128	0.271^3	5
0019	1.9768	0	0	1	1	150	Eq. 2	128× 128	0.271^3	5
0020	1.9768	0	0	1	1	200	Eq. 2	128× 128	0.271^3	5
0021	1.9768	0	0	1	1	250	Eq. 2	128× 128	0.271^3	5
0022	1.9768	0	0	1	1	300	Eq. 2	128× 128	0.271^3	5
0023	1.9768	0	0	1	1	400	Eq. 2	128× 128	0.271^3	5
0024	1.9768	0	0	1	1	500	Eq. 2	128× 128	0.271^3	5

Table 2: The values of the parameters used for the runs. The total etching time is 120 s for all cases. A_{SiCl_x} and A_{Si} are the parameters defining the ion-enhanced and physical sputtering yield (equation 1)), a_{chem} is the coefficient of pure chemical reaction of a saturated (with Cl) Si surface, x_+ is the fraction of ions in the arriving flux, S_d is the sticking probability of the etching products, E_+ is the ion energy, $f_{SiCl_x}(\theta)$ is the function defining the angle dependence of the ion-enhanced etching yield (equation 2))

3.2 First Comparison of the Calculated Etching Rate with Measurements

The first comparison of the model results with the measurements is made at an ion energy of 100 eV where the etching rate was measured [32] ~260 nm/min (at 120 s), respectively. When we use the values proposed by Tsuda et al. [31, 34] in the surface model (see case 0002 in Table 2), the etching rate is calculated ~60 nm/min, i.e. four times smaller than the experimental value. In order to investigate the origin of the difference between the model and the experimental results, the first step was to check the effect of the model assumptions on the results. After some trial runs, it was found that, if the angle dependence of the ion enhanced etching yield ($f_{SiCl_x}(\theta)$) and the product redeposition were both neglected, the etching rate was calculated ~270 nm/min, i.e. very close to the experimental results. If we accept that the angle dependence of the etching yield is a fact (and thus we cannot assume that $f_{SiCl_x}(\theta)$=1), and given that the dominant mechanism of Si etching by Cl_2 plasma is ion-enhanced etching, a critical parameter for the value of the calculated etching rate is A_{SiCl_x} of equation 1. A "nanoscopic" value of the A_{SiCl_x} is required for the MC framework. The procedure we followed to extract this "nanoscopic" value is described in section 3.3.

3.3 Extraction of "Nanoscopic" Values of the Parameter Defining the Ion-enhanced Etching Yield

The "nanoscopic" value of A_{SiCl_x} (equation 1) is extracted by a trial and error procedure: Several values of A_{SiCl_x} are checked until the calculated etching rate at an ion energy equal to 100 eV is close to the measured value of the etching rate (~270 nm/min). The uncertainty of the value of the sticking probability of the redeposited products, S_d (S_d=0 means no redeposition of etching products, S_d=1 means that all molecules of etching products stick with 100% probability), manifests the requirement for a parametric analysis: The "nanoscopic" value of A_{SiCl_x} reproducing the experimentally measured etching rate will be calculated by the trial and error procedure for several values of S_d from 0 to 1. The "nanoscopic" values of A_{SiCl_x} are shown in Table 3.

case/ S_d	0.0	0.001	0.005	0.01	0.05	0.1	0.2	0.5	1.0
0009	0.3354	0.4766	0.5648	0.6001	0.6354	0.6354	0.6707	0.6884	0.7237
0010	1.3061	1.6238	2.0121	1.9768	1.9768	1.8356	1.6944	1.4197	1.3061
0011		1.6238	1.9415	1.9768	1.9768	1.8356	1.6591	1.4197	1.3061
0012			1.9415	1.9768	1.9768	1.8356	1.6591	1.4197	1.3061

Table 3: The "nanoscopic" values of A_{SiCl_x} (in $eV^{-0.5}$) and S_d reproducing the experimental value of the etching rate for an ion energy equal to 100 eV (~260 nm/min) and for different values of S_d. The value proposed by Tsuda et al. [32] is 0.353 $eV^{-0.5}$

3.4 First Comparison of the Calculated Etching Rate with Measurements

The extraction of "nanoscopic" value of A_{SiCl_x}, as described in the section 3.2, was performed at an ion energy equal to 100 eV. But does this "nanoscopic" value predict the behavior of etching rate and rms roughness for other values of ion energy? The answer is included in figure 3 where the simulation results are shown versus the ion energy. Two values of S_d (sticking probability of etching products) are used, i.e. 0.05 and 1. All runs included in figure 3 are described by cases 0009-0016 (S_d=0.05) and 0017-0024 (S_d=1) in Table 2. In figure 3, the measured [14] etching rate and rms roughness are also shown. The MC modeling framework reproduces well the experimentally measured dependence of the etching rate on the ion energy (see figure 3a) for both values of S_d (0.05 and 1). Greater deviations are observed at greater ion energies, i.e. above 350 eV. Regarding the rms roughness (figure 3b), although the absolute values are overestimated, the behavior of rms roughness versus ion energy is captured by the MC modeling framework for the case S_d=1. It is interesting that the framework captures the existence of a maximum of rms roughness versus the ion energy. The maximum of the rms roughness is at ~300 eV for the simulation results and at ~250 eV for the measurements. When S_d=0, the calculated rms values are much greater compared to the measured values. The calculated values are getting lower when S_d=1, i.e. when the redeposition of the etching products is more intense. The redeposition favors lower values for rms. When redeposition is less intense, there is no maximum or saturation of the rms with the ion energy (see figure 3b, S_d=0). The potential origins of the deviations between the simulation results and the experimental data are the following.

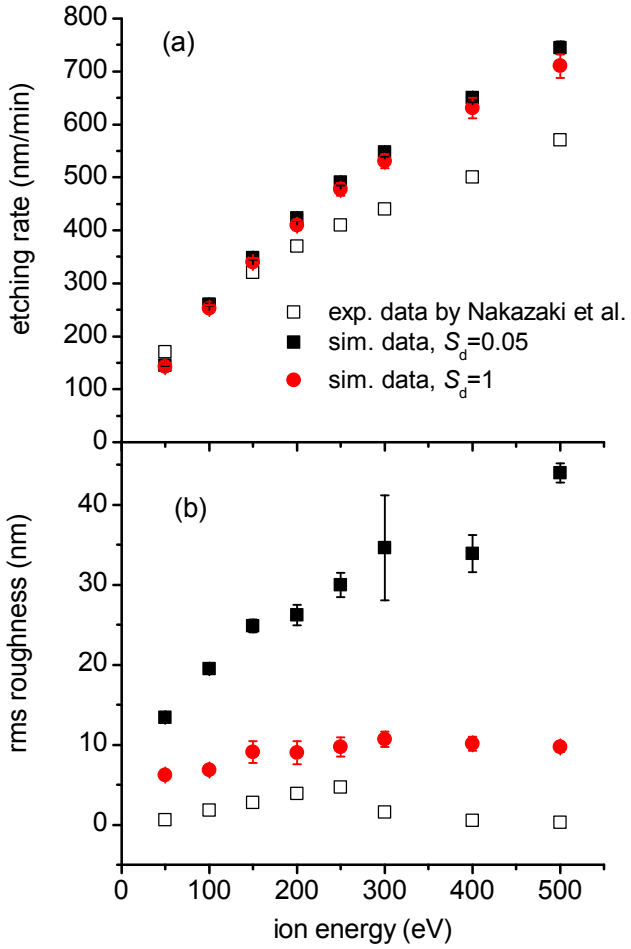


Figure 3: a) The etching rate and b) the rms roughness vs the ion energy as obtained from our simulation and the measurements of Nakazaki et al. [20]. The simulation results refer to cases 0014 to 0029: Two values of S_d are considered: S_d=0.05 and S_d=1. The error bars are coming from the standard deviations of the runs of each case

First, regarding the overestimation of the etching rate at high ion energies by the MC modeling framework, the assumption of a fully chlorinated surface may not be valid at high ion energies; if the chlorination level of the surface (x in equation 1) decreases with the ion energy, the etching rate will also decrease. Additionally, the composition of the flux arriving on the surface may also change as the ion energy increases. The production rate of etching products increases with the ion energy and these etching products join the plasma reactions and may alter the composition of ions arriving on the surface. Tsuda et al. [32] reported that the fraction of heavy (light) ions, e.g. $SiCl_x^+$ (Cl^+), on the arriving flux increases (decreases) with the ion energy. The ion-enhanced etching yield of Si by $SiCl_x^+$ may be lower compared to the ion-enhanced yield of Si by Cl^+. **Second**, regarding the overestimation of the rms roughness by the MC modeling framework, there may be an underestimation of the sticking probability of ions on the surface. The reduction of the ion reflection is expected to decrease the rms roughness; ion reflection is a mechanism which enhances

rms roughness. Additionally, given that the sticking probability of ions depends on the angle of ion incidence, ion reflection may be also reduced by changing the method for the calculation of the local slope of the surface. In the MC modeling framework, the local slope of the surface at an impact point (point where an ion arrives) is calculated by exploiting the values of the first neighbors of the impact point. The use of more neighbors would decrease the local slope and thus increase the sticking probability of the ions. The local slope has been calculated in previous works by taking into account 4 adjacent neighbors [24] and by taking into account 125 neighbors [31]. Regardless of the available options for the calculation of the local slope, the question remains. The right choice requires a new study to investigate the notion of local slope in MC calculations.

4. CONCLUSIONS & FUTURE WORK

A surface model for Si etching by Cl_2 plasma was developed and it was coupled with a MC modeling framework to predict the etching rate and the rms roughness of Si surfaces.

When the values of the surface model parameters were taken from the literature, the etching rate calculated by the MC framework was 4 times lower compared to measurements. These values are usually coming from fitting to measurements and thus they inherently entail an average effect of the angle of ion incidence. This is the “macroscopic” etching yield which captures the net effect of the ion bombardment taking into account all surface processes (including redeposition and angle dependence of etching) but is not suitable for a MC framework which treats each surface process separately.

For a MC framework, the “nanoscopic” etching yield is required suitable to reproduce the “macroscopic” etching yield and rate. This is the procedure followed in this work: The “nanoscopic” etching yield of the dominant etching mechanism, namely ion-enhanced etching, was extracted by fitting the calculated etching rate to available measurements. This procedure was performed for one value of the ion energy (100 eV), and then the results of the modeling framework were compared to measurements of etching rate and rms roughness for different values of the ion energy (50 to 500 eV).

The MC framework reproduced well the experimentally measured dependence of the etching rate on the ion energy. The etching rate was overestimated at ion energies greater than 350 eV.

Regarding the rms roughness, although the absolute values are overestimated, the behavior of rms roughness versus ion energy is captured when the redeposition of the etching products is intense. Besides the comparison with measurements, the simulations with the MC framework showed that coarse graining affected the value of the rms value being calculated, thus it was avoided.

Regarding the future works, given that the local slope is critical for both ion reflection and the ion-enhanced etching yield, a new investigation for right treatment of local slope will be a useful extension of the present study. Additionally, a study on the origin of the angle dependence of the etching yield and the means to incorporate this dependence in a MC framework will be very interesting. Finally, the surface model and the MC modeling framework can be integrated in a multiscale modeling framework [38] to take into account the change of the ion composition of the arriving flux.

REFERENCES

[1] Friedrich L. Bauer. “Origins and Foundations of Computing”. *Springer-Verlag Berlin Heidelberg*. 2010.

[2] J. D. Plummer, M. Deal, and P. B. Griffin. “Silicon VLSI Technology. Fundamentals, Practice and Modeling”. *New Jersey: Prentice Hall.* 2000.

[3] M. A. Lieberman and A. J. Lichtenberg. “Principles of Plasma Discharges and Materials Processing”. *John Wiley & Sons, Inc., Hoboken, New Jersey.* 2005.

[4] W. Guo and H. H. Sawin. "Review of profile and roughening simulation in microelectronics plasma etching". *Journal Of Physics D-Applied Physics,* vol. 42, no. 19, p. 194014. 2009.

[5] N. Vourdas, A. Tserepi, and E. Gogolides. "Nanotextured super-hydrophobic transparent poly(methyl methacrylate) surfaces using high-density plasma processing". *Nanotechnology,* vol. 18, no. 12, p. 125304. 2007.

[6] E. Gogolides et al. "Controlling roughness: from etching to nanotexturing and plasma-directed organization on organic and inorganic materials." *Journal Of Physics D: Applied Physics,* vol. 44, no. 17, p. 174021. 2011.

[7] K. Tsougeni, A. Tserepi, V. Constantoudis, E. Gogolides, P. S. Petrou, and S. E. Kakabakos, "Plasma Nanotextured PMMA Surfaces for Protein Arrays: Increased Protein Binding and Enhanced Detection Sensitivity". *Langmuir,* vol. 26, no. 17, pp. 13883-13891. 2010.

[8] J. Shen and J. Kirschenr, "Tailoring magnetism in artificially structured materials: the new frontier." *Surface Science,* vol. 500, no. 1-3, pp. 300-322. 2002.

[9] G. Costantini *et al.*, "Tuning surface reactivity by in situ surface nanostructuring." *Journal Of Chemical Physics,* vol. 112, no. 15, pp. 6840-6843. 2000.

[10] N. Vourdas *et al.*, "Plasma directed assembly and organization: bottom-up nanopatterning using top-down technology". *Nanotechnology,* vol. 21, no. 8, p. 085302. 2010.

[11] J. W. Coburn and H. F. Winters, "Ion- and electron-assisted gas-surface chemistry. An important effect in plasma etching". *Journal Of Applied Physics,* vol. 50, no. 5, p. 3189. 1979.

[12] G. Kokkoris and E. Gogolides, "The potential of ion-driven etching with simultaneous deposition of impurities for inducing periodic dots on surfaces." *Journal of Physics D: Applied Physics,* vol. 45, no. 16, p. 165204. 2012.

[13] G. Kokkoris, V. Constantoudis, P. Angelikopoulos, G. Boulousis, and E. Gogolides, "Dual nanoscale roughness on plasma-etched Si surfaces: Role of etch inhibitors". *Physical Review B - Condensed Matter and Materials Physics,* vol. 76, no. 19, p. 193405. 2007.

[14] N. Nakazaki, H. Tsuda, Y. Takao, K. Eriguchi, and K. Ono, "Two modes of surface roughening during plasma etching of silicon: Role of ionized etch products". *Journal of Applied Physics,* vol. 116, no. 22, p. 223302. 2014.

[15] R. Petri *et al.*, "Silicon Roughness Induced By Plasma-Etching". *Journal Of Applied Physics,* vol. 75, no. 11, pp. 7498-7506. 1994.

[16] W. S. Hwang, B. J. Cho, D. S. H. Chan, S. W. Lee, and W. J. Yoo, "Effects of volatility of etch by-products on surface roughness during etching of metal gates in Cl_2". *Journal Of The Electrochemical Society,* vol. 155, no. 1, pp. H6-H10. 2008.

[17] S. Kuo-Tung and W. P. Stella "Mass Spectrometry, Optical Emission Spectroscopy, and Atomic Force Microscopy Studies of Si Etch Characteristics in a C l 2 Plasma Generated by an Electron Cyclotron Resonance Source". *Japanese Journal of Applied Physics,* vol. 33, no. 12S, p. 7112. 1994.

[18] G. Boulousis, V. Constantoudis, G. Kokkoris, and E. Gogolides, "Formation and metrology of dual scale nano-morphology on SF6 plasma etched silicon surfaces". *Nanotechnology,* vol. 19, no. 25, p. 255301. 2008.

[19] Y. Yin and H. H. Sawin, "Surface roughening of silicon, thermal silicon dioxide, and low- k dielectric coral films in argon plasma". *Journal of Vacuum Science and Technology A: Vacuum, Surfaces and Films,* vol. 26, no. 1, pp. 151-160. 2008.

[20] M. Martin and G. Cunge, "Surface roughness generated by plasma etching processes of silicon". *Journal of vacuum Science & Technology B,* vol. 26, no. 4, pp. 1281-1288. 2008.

[21] J. A. Levinson, E. S. G. Shaqfeh, M. Balooch, and A. V. Hamza, "Ion-assisted etching and profile development of silicon in molecular and atomic chlorine". *Journal Of Vacuum Science & Technology B,* vol. 18, no. 1, pp. 172-190. 2000.

[22] E. S. G. Shaqfeh and C. W. Jurgensen, "Simulation Of Reactive Ion Etching Pattern Transfer". *Journal Of Applied Physics,* vol. 66, no. 10, pp. 4664-4675. 1989.

[23] G. Kokkoris, A. Tserepi, A. G. Boudouvis, and E. Gogolides, "Simulation of SiO2 and Si feature etching for microelectronics and microelectromechanical systems fabrication: A combined simulator coupling modules of surface etching, local flux calculation, and profile evolution". *Journal of Vacuum Science and Technology A,* vol. 22, no. 4, pp. 1896-1902. 2004.

[24] Y. Osano and K. Ono, "An atomic scale model of multilayer surface reactions and the feature profile evolution during plasma etching". *Japanese Journal Of Applied Physics Part 1-Regular Papers Brief Communications & Review Papers,* Article vol. 44, no. 12, p. 8650. 2005.

[25] Y. Zhang, C. Huard, S. Sriraman, J. Belen, A. Paterson, and M. J. Kushner, "Investigation of feature orientation and consequences of ion tilting during plasma etching with a three-dimensional feature profile simulator". *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films,* vol. 35, no. 2, p. 021303. 2017.

[26] W. Guo, B. Bai, and H. H. Sawin, "Mixing-layer kinetics model for plasma etching and the cellular realization in three-dimensional profile simulator". *Journal Of Vacuum Science & Technology A,* vol. 27, no. 2, pp. 388-403. 2009.

[27] A. Sankaran and M. J. Kushner, "Integrated feature scale modeling of plasma processing of porous and solid SiO2. I. Fluorocarbon etching," *Journal of Vacuum Science and Technology A: Vacuum, Surfaces and Films,* vol. 22, no. 4, p. 1242. 2004.

[28] M. Wang and M. J. Kushner, "High energy electron fluxes in dc-augmented capacitively coupled plasmas. II. Effects on twisting in high aspect ratio etching of dielectrics". *Journal Of Applied Physics,* vol. 107, no. 2, p. 023309. 2010.

[29] C. M. Huard, Y. Zhang, S. Sriraman, A. Paterson, K. J. Kanarik, and M. J. Kushner, "Atomic layer etching of 3D structures in silicon: Self-limiting and nonideal reactions," *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films,* vol. 35, no. 3, p. 031306. 2017.

[30] J. T. Drotar, Y. P. Zhao, T. M. Lu, and G. C. Wang, "Surface roughening in shadowing growth and etching in 2+1 dimensions". *Physical Review B,* vol. 62, no. 3, pp. 2118-2125. 2000.

[31] H. Tsuda, H. Miyata, Y. Takao, K. Eriguchi, and K. Ono, "Three-Dimensional Atomic-Scale Cellular Model and Feature Profile Evolution during Si Etching in Chlorine-Based Plasmas: Analysis of Profile Anomalies and Surface Roughness". *Japanese Journal Of Applied Physics,* vol. 50, no. 8, p. 08JE06. 2011.

[32] H. Tsuda, N. Nakazaki, Y. Takao, K. Eriguchi, and K. Ono, "Surface roughening and rippling during plasma etching of silicon: Numerical investigations and a comparison with experiments". *Journal of Vacuum Science and Technology B: Microelectronics and Nanometer Structures,* vol. 32, no. 3, p. 031212, 2014, Art. no. 031212.

[33] E. A. Ogryzlo, D. E. Ibbotson, D. L. Flamm, and J. A. Mucha, "Doping and crystallographic effects in Cl-atom etching of silicon". *Journal of Applied Physics,* vol. 67, no. 6, p. 3115. 1990.

[34] N. Nakazaki, H. Matsumoto, H. Tsuda, Y. Takao, K. Eriguchi, and K. Ono, "Surface smoothing during plasma etching of Si in Cl2". *Applied Physics Letters,* vol. 109, no. 20, p. 204101. 2016.

[35] W. Guo and H. H. Sawin, "Modeling of the angular dependence of plasma etching". *Journal Of Vacuum Science & Technology A,* vol. 27, no. 6, pp. 1326-1336. 2009.

[36] H. Tsuda, Y. Takao, K. Eriguchi, and K. Ono, "Modeling and simulation of nanoscale surface rippling during plasma etching of Si under oblique ion incidence". *Japanese Journal of Applied Physics,* vol. 51, no. 8 PART 2, 2012, Art. no. 08hc01.

[37] A. L. Barabasi and H. E. Stanley, “Fractal Concepts in Surface Growth”. Cambridge University Press. 1995.

[38] S. Mouchtouris and G. Kokkoris, "Multiscale Modeling of Low Pressure Plasma Etching Processes: Linking the Operating Parameters of the Plasma Reactor with Surface Roughness Evolution". *Plasma Processes and Polymers,* vol. 14, p. 1600147. 2017.

Advanced clustering methods for identifying bioactive molecular conformations

Christoforou Emmanouil (echristo@di.uoa.gr, emchristoforou@gmail.com)

ABSTRACT

Molecular Dynamics simulations is a powerful technique for studying the structure and dynamics of biomolecules in atomic-level detail. Due to the long timescales and the big and complex nature of the corresponding data, relevant analyses of important biophysical phenomena are challenging. Clustering and Markov State Models are efficient computational techniques that can be used to extract dominant conformational states and to connect them with kinetic information. We investigate the free energy landscape of Angiotensin II (AII) in order to unravel its bioactive conformation, using multiple trajectories that are analysed with clustering techniques and Markov State modeling.

Our results show that in the water-ethanol environment Angiotensin II adopts more compact U-shaped (folded) conformations than in water, which resembles its structure when bound to the AT1 receptor, a transmembrane protein. For clustering of the conformations we validate the efficiency of an inverted-quantized k-means algorithm (IQ-means), as a fast approximate clustering technique compared to k-means with reasonable trade-offs between time and accuracy. Finally, we extract Markov State Models using various clustering techniques to generate microstates and macrostates, as well as to select macrostate representatives.

Keywords: Clustering, Markov State Models, Molecular Dynamics Simulations, Approximate Clustering

Advisors
Ioannis Z. Emiris, Professor (NKUA), Zoe Cournia, Researcher (BRFAA)

1. INTRODUCTION

Molecular Dynamics (MD) simulation is a computer simulation method to study the physical movements of atoms and molecules over a given period of time, observing the evolution of the system. Due to their atomic-level detail, MD simulations can describe the dynamics of molecules, providing a much higher temporal and spatial resolution than most experimental techniques. This allows for an efficient study of the conformational ensemble of proteins, which consists an essential part of many biological processes such as protein folding and ligand binding.

Due to the long timescales that need to be sampled and the big and complex nature of the corresponding data, relevant analyses of important biophysical phenomena is challenging. Clustering and Markov State Models are efficient computational techniques that can be used to extract dominant conformational states and to connect those with kinetic information.

For long simulations resulting in big datasets, approximate but efficient clustering techniques could be employed to replace common clustering methods such as k-means in order to reduce the computational effort. In this work, we validated the efficiency of an inverted-quantized k-means algorithm (IQ-means) [1] on data from trajectories of MD simulations, as a fast approximate clustering technique compared to k-means, with reasonable trade-offs between time and accuracy. IQ-means, using multiple ingredients from advanced approximate k-means variants, has been reported to achieve clustering of 100 million images on a single machine in less than an hour. Therefore, its application could be significant not only as an approximation to replace k-means, but also as a step to determine the number of clusters in a dataset, before using more accurate and time-consuming techniques.

In addition to the need for fast clustering solutions, another important aspect to efficiently analyse long trajectories is to produce simple meaningful representations that allow straightforward visualizations, as well as faster processing and transfer of the trajectories. A metadata representation, where each structure is converted into a single 3D point using the eigenvalues from the Multidimensional Scaling of the backbone atoms distances was validated in this work. The resulting eigenvalues appear to provide simple representations reproducing the different shapes of the conformations.
To extract the long-time statistical conformational dynamics from the MD simulations, Markov State Models (MSMs) [2] were generated using various clustering techniques. MSMs can predict both stationary and kinetic quantities on long timescales (e.g. milliseconds) using a set of atomistic MD simulations. Here we also validated IQ-means as an approximation technique for the generation of the microstates in MSMs. For the representatives of each macrostate we chose the centroids with the largest neighborhood from GROMOS clustering.

These methods were applied to investigate the bioactive conformation of the octapeptide Angiotensin II (AII), using multiple trajectories from simulations in

mixtures of water and water-ethanol at various temperatures. There have been several works for the bioactive conformation of Angiotensin II when bound to the AT1 receptor. However, some studies show that Angiotensin II adopts a compact folded (U-shaped) structure, others indicate an extended β-structure, while others indicate a mixture of unfolded and folded conformations. Also, various studies suggest that organic solvents and heating appear to favor the folding of the Angiotensin amide into a compact structure. These diverging results could be explained by the flexibility of Angiotensin II, the different solvents used for each experiment and the different experimental conditions. The simulations used in this work were performed in water and water-ethanol mixtures to mimic the water-membrane interface as AII approaches the AT1 receptor.

Our results show that in water-ethanol environment Angiotensin II adopts more compact folded (U-shaped) conformations than in water, which resembles its structure when bound to the AT1 receptor. Also, we show that the eigenvalues of the Multidimensional Scaling on the backbone atoms distances could provide a meaningful metadata representation. Finally, we extract Markov State Models using various clustering techniques to generate microstates and macrostates, as well as to select macrostate representatives. IQ-means appeared to perform well on data from conformations of trajectories, as a very fast approximation with a fair loss in quality and also as an approximate clustering solution to determine the microstates in MSMs.

The rest of the paper is organized as follows. Section 2 outlines the methods used to identify the bioactive conformation of Angiotensin II. Section 3 reveals the results of the clustering techniques and Markov State models. Finally, Section 4 concludes the work.

2. METHODS

In this section we describe the methods that were used to identify the bioactive conformation of Angiotensin II. We describe Angiotensin's II MD simulations, as well as the clustering methods, the Markov State Models and the metadata representations that were used to analyse the results of the simulations.

2.1 Molecular Dynamics (MD) simulations

MD simulation is a computer simulation method to study the physical movements of atoms and molecules over a given period of time, observing the evolution of the system. They are performed to understand the properties of molecules, such as their structure, as well as the interactions between them. For a system of interacting particles, we begin with an initial structure usually derived from experimental techniques, and we set topology and force fields. The trajectories of atoms and molecules are calculated by numerically solving Newton's equations of motion.

The MD simulations that are used in this work were accomplished by Dr. Hari Leontiadou and Dr. Zoe Cournia. To investigate the bioactive conformation of Angiotensin II our work focuses in an atomic-level picture of the intermolecular interactions of Angiotensin II in water and water/ethanol mixtures. These solvents are selected since Angiotensin II binds to the transmembrane protein AT1 and the water/ethanol mixtures are known to mimic the membrane environment. Different temperatures (278K, 298K, 310K, 323K) around the average human temperature were used in order to verify if heating affects the structures of the conformations. The initial structure used for the octapeptide Angiotensin II is 1N9V.pdb from Protein Data Bank [3].

Furthermore, to increase the sampling for each system, ten different conformations were used as starting positions to initiate ten independent simulations, each one lasting 100ns. Conformations were saved every 10ps, therefore for each system we have simulations of 1μs (10 simulations of 100ns). Thus, each simulation consists of 100000 conformations.
We compare the conformations in water and water/ethanol mixtures to investigate if Angiotensin II achieves a compact U-shaped (folded) structure or an extended β-structure (unfolded), when it interacts with the membrane environment.

2.2 Clustering

2.2.1 *k*-means

The k-means clustering algorithm is one of the most popular and well-studied algorithms. Its purpose is to partition the data into k number of clusters, where each observation is assigned to the cluster with the closest center (mean point). Initialization with k-means++ provides a faster approach for k-means, since it converges in only a few iterations, having reproducible results.

2.2.2 Inverted-Quantized k-means

Inverted-quantized k-means (IQ-means) [1] is a method for approximate clustering, using multiple ingredients from advanced approximate *k*-means variants. Among others, some key concepts of IQ-means are the fine representation of data in a 2d grid, a multi-index based inverted search from centroids to cells and a dynamic version of the algorithm that comes as a natural extension from EGM. Combined with efficient deep learning representation it has been reported to achieve the clustering of 100 million images on a single machine in less than an hour.

For the representation of the data, IQ-means begins by learning a codebook as in the inverted multi-index, using a sample from the data that are about to be clustered. Alternatively, one could use an already pre-trained codebook. Then all points are quantized on the grid, like DRVQ and a discrete two-dimensional distribution p of points over cells is constructed. Finally, the algorithm alternates between an assignment and an update step, as a proper k-means variant.

The assignment step, in order to achieve fast searching, takes the form of searching for a set of individual queries in the nearest cells, one for each centroid. The search follows the multi-index approach of the multi-sequence algorithm. During the search process, the algorithm instead of searching for the nearest centroid of each cell, follows a reverse approach where for each centroid looks for the nearest cells (points) using a window.

2.2.3 GROMOS

Gromos is a method used for clustering structures [4], addressing the issue of finding clusters (groups) of structures in a dataset given a distance cut-off. The structures are clustered by comparing their root-mean-square deviation of atomic positions (RMSD) values, which is the average distance between their atoms, usually of the backbone or the Calpha atoms. More specific, GROMOS finds the structure with the largest neighbourhood according to the distance cut-off and defines it as the first cluster medoid. Then, this structure and its neighbours are eliminated from the pool of conformations and the algorithm iterates until all structures are assigned to a cluster.

2.3 Markov State Models

Markov state models are discrete-time models based on the kinetic exchange between states, which describe a decomposition of the conformational space into small metastable regions [11,12]. They provide the means to efficiently understand and gain an insight from simulation data with complex nature, by predicting long timescale dynamics from long, or multiple short, trajectories.

The construction of a Markov State Model is far from trivia, since it involves a lot of decisions. The key steps to build an MSM are:

[1] The selection of features from the MD trajectories and the application of TICA transformation, a dimension reduction technique, to prepare the state-space.
[2] The “geometric” clustering step to discretize the trajectories into finite states, the microstates.
[3] The estimation of a transition probability matrix for the microstates choosing an appropriate lag time for the Markov model.
[4] The “kinetic” clustering step, to group the microstates by the transition probability matrix into sets of kinetically related states, the macrostates.
[5] The coarse-graining of the kinetic model based on the produced macrostates.

Feature selection and Dimension Reduction. Instead of raw MD trajectory conformations, we use a set of features selected in order to characterize best the rare-event transitions. Here we keep all the dihedral angles (all backbone phi/psi and chi1 angles) and the minimum distances between peptide heavy atoms, which are usually best suited for small peptides, such as Angiotensin II. The resulting dataset consists of 59 features.

For the efficient generation of the microstates it is recommended to reduce the dimension of the selected features in order to improve the quality of the discretizations and the CPU time needed [2]. **Time-lagged independent component analysis (TICA)** is a linear transformation method which finds coordinates of maximal autocorrelation at a given lag time and has been shown that is optimal in approximating the relevant slow reaction coordinates from MD simulations. Thus, TICA is considered to be ideal to construct Markov State Models. The dimension of our data is reduced by TICA to 26, preserving 95% of the kinetic variance.

Microstates. For the discretization of the conformational space clustering techniques such as k-means, k-medoids, GROMOS, etc can be used. Here, a k-means clustering with 100 clusters was carried out. We also compare the k-means microstates with those generated by IQ-means.

Estimation of Markov State Models. Markov State Models are the models that describe the kinetics of molecules with a matrix of conditional transition probabilities among the microstates. The model is composed of the conditional probabilities for a state space that consists of s(t) discrete trajectories, jumping between n microstates at lag time τ:

$$p_{ij} = Pr(s(t+\tau) = j \vee s(t) = i)$$

The probability of jumping from state j starting from i, is computed by the maximum likelihood estimator:

$$Pr\big(C(\tau) \vee P(\tau)\big) \propto \prod_{i,j=1}^{n} p_{ij}^{c_{ij}(\tau)}$$

Implied timescales. To ensure the accuracy of the MSM we select a lag time, so that the implied relaxation timescales are approximate constant within the statistical error. The behavior of the implied timescales consists a way to check if the model is Markovian.

Implied timescales refer to the relaxation timescales of a molecule implied by the transition matrix of a Markov model at a lag time τ and is given by$t_i(t) = -\frac{\tau}{ln|\lambda_i(\tau)|}$, where $\lambda_i(\tau)$ is the i-th eigenvalue of the transition matrix $P(\tau)$. For a Markovian system $\lambda_i(\tau)$ should be constant and independent of the lag time τ.

In this work, it seems that in all cases (AII simulations) the converged time scales are observed at a lag time τ=200steps=2ns. Therefore, a lag time τ=2ns was chosen for the construction and coarse-graining of the MSM models.

Model Validation with Chapman-Kolmogorov Test. The model at the selected lag time (here τ=2ns) is validated using Chapman-Kolmogorov test, a formulation for the Chapman-Kolmogorov equation ($P(k\tau) = P^k(\tau)$). Due to this test, a Markov model estimated at a lag time τ, should be able to predict estimates performed at longer timescales $k\tau$ within the statistical error.

Macrostates. The model is coarse-grained using the transition probabilities among the microstates to a simpler, “humanly” readable, model. There is a variety of coarse-graining techniques (PCCA [5], BACE [6]) to simplify the

model, exploiting the kinetic relevance of the states. Here, we performed Perron cluster cluster analysis (PCCA++[7]) with 8 macrostates.

Perron Cluster Cluster Analysis (PCCA) [5] is one of the most common methods used for coarse-graining Markov State Models, exploiting the eigenspectrum of the transition probability matrix. PCCA begins with all the microstates merged into one, big, coarse-grained, macrostate and then it iteratively splits the most kinetically diverse macrostate into two states, until the requested number of states is reached. The most common approaches to find the meta-stable states of Markov State Models are PCCA+ [8] and PCCA++ [7].

Kinetic Modeling. The final coarse-grained kinetic model and the estimation of the transition rates between the metastable states are generated using Hidden Markov Models (HMM). HMMs consist an efficient approximation of the kinetics on discretized molecular state spaces [9].

Representatives. In this work the macrostates representatives are selected using the GROMOS [4] clustering algorithm. Instead of randomly sampling from the macrostates, GROMOS is applied on large groups of sampled conformations from each macrostate and the medoid of the bigger cluster is assumed as a "dominant" representative.

2.4 Metadata representatives

Metadata representations for protein conformations in a trajectory, as suggested by Zhang et al. [10], are simple 3D points representatives produced by Multidimensional Scaling (MDS). To generate these representations a distance matrix of the backbone atoms is computed for each conformation and then classical MDS is applied to reduce the dimension of the matrix to 3. The eigenvalues generated by MDS seem to preserve the variations in the data, consisting a fine metadata representation with a single 3D point for each conformation. Such representations allow us to efficiently visualize the simulations into simple plots.

3. RESULTS

At this section we exhibit the analysis of the simulations, the comparison of k-means with IQ-means and the results of the MSMs, build by PyEMMA2 [2].

3.1 Angiotensin II

Angiotensin II is an octapeptide (Asp-Arg-Val-Tyr-Ile-His-Pro-Phe) with 147 atoms. When folded it appears to obtain a compact U-shaped structure. Otherwise, it obtains an extended (E) or intermediate (I) structure. In the following figure (Fig. 1) we observe Angiotensin II and its key structures (U,I,E) (Fig. 1b,c,d) represented in cartoon. To identify the various structures we use the end-to-end distances, which are the distances of the C- and N-terminal residues (ASP-PHE). Smaller distances indicate a folded state (U-shaped

structure), while greater distances indicate intermediate (I) or fully extended (E) structures.

We compare the conformations in water and water/ethanol mixtures to investigate whether Angiotensin II achieves a compact U-shaped (folded) structure or an extended β-structure (unfolded), when interacting with the membrane environment. The histograms and probability density plots for the end-to-end distances of the simulated systems (Fig. 2) indicate that All conformations concentrate mainly in 3 peaks: U (u-shaped)=0.5nm, I (intermediate)=1.1-1.3nm and E (extended)=1.8-2.0nm. It appears that in the presence of ethanol molecules in the solvent the peptide adopts more U-shaped structures than in the water solvent.

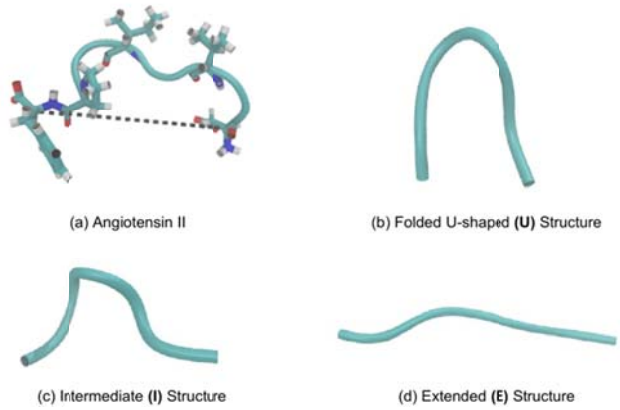


Figure 1. Angiotensin's II structure and shapes. (a) Dotted line indicates the end-to-end distance of terminal residues. Conformations are represented in cartoon, visualized by VMD.

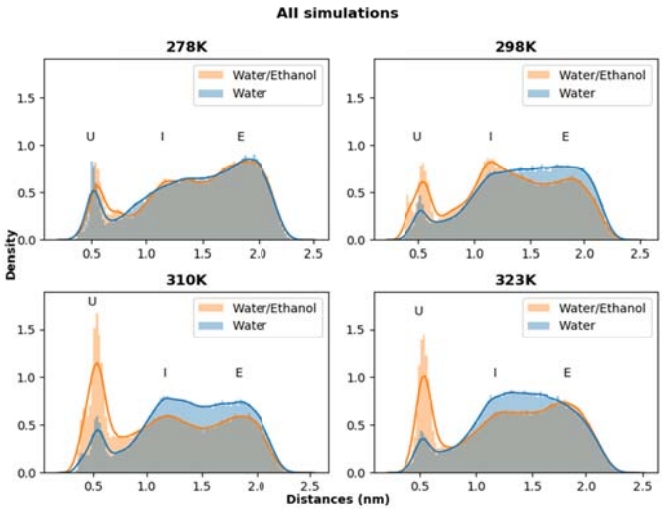


Figure 2. Histogram and Probability Densities (P(r)) of end-to-end distances of the terminal residues in water (blue) and Water/Ethanol (somon). Conformations concentrate mainly in 3 peaks: U (u-shaped)=0.5nm, I (intermediate)=1.1-1.3nm and E (extended)=1.8-2.0nm.

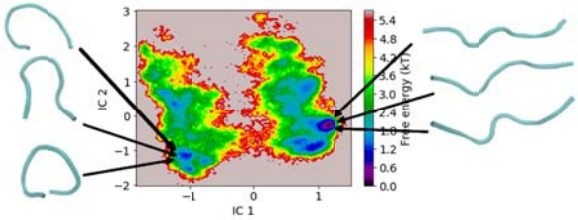
3.2 Markov State Models

We build MSMs for Angiotensin II at 310K, which is the most interesting temperature as it is closer to the conditions faced by Angiotensin's II cells in the human body.

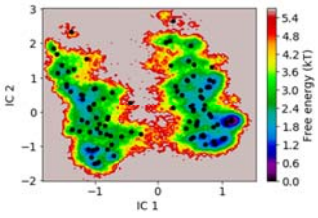
3.2.1 MSM for All using k-means for the generation of the microstates

MSM for All in Water/Ethanol at T=310K

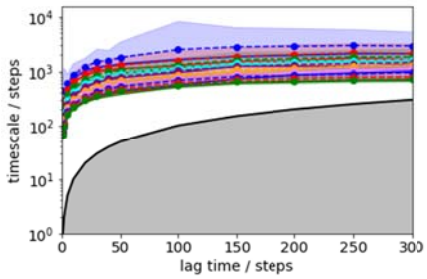
To construct the MSM we select all the dihedral angles and the minimum distances between peptide heavy atoms, resulting to a dataset of 59 features. Using TICA we reduce the dimension of our data to 26 preserving 95% of the kinetic variance (Fig. 3a) and we discretize our trajectories using k-means with 100 centers (Fig. 3b). The converged time scales are observed at a lag time τ=200steps=2ns (Fig. 3c,d).



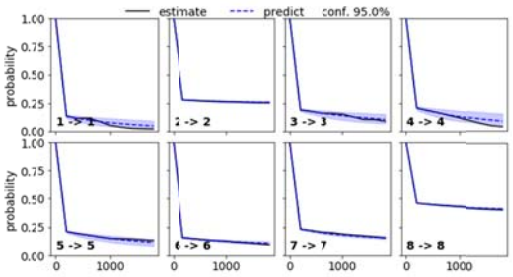
(a) Free energy plot of the space created by the first two TICA components (IC1, IC2)



(b) k-means with 100 cluster centers on TICA coordinates



(c) Implied timescales with Error bars



(d) Chapman-Kolmogorov test with lag time 200 steps

Figure 3. MSM for All in Water/Ethanol at T=310K.

Finally, after validating the model, the microstates are coarse-grained at a lag time τ=2ns using PCCA++ for the macrostates and HMM for their kinetic model (Fig. 4). Representatives for each state are selected using GROMOS with a cut-off of 0.15nm (Fig. 4b). For each metastable state there is a corresponding probability density graph that indicates the structural preference in the specific metastable state (Fig. 4a).

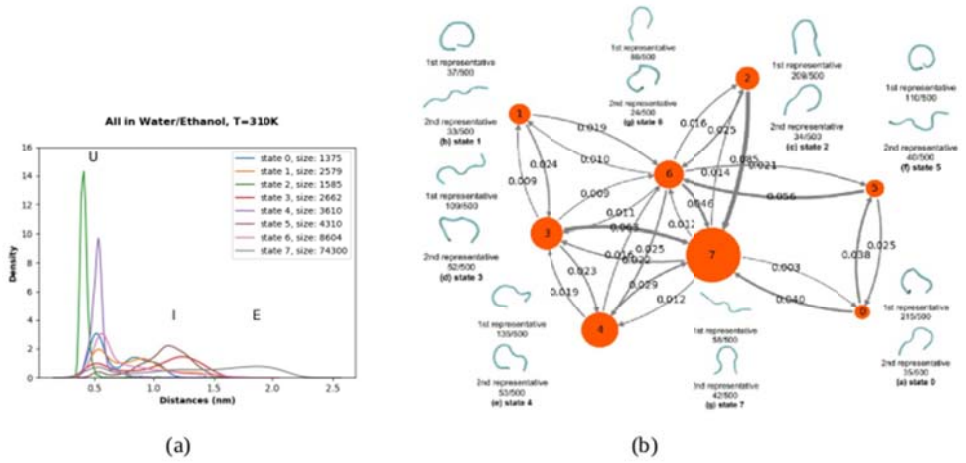


Figure 4. MSM for All in Water/Ethanol at 310K. (a) Probability Density plots of end-to-end distances for each macrostate. (b) Kinetic model of MSM with representatives for each state using GROMOS.

MSM for All in Water at T=310K

Likewise, we construct the MSM for Angiotensin II in Water (Fig. 5). We use the same features as in the previous MSM, we discretize the trajectories using 100 microstates, and we choose a lag time at τ=200steps=2ns and 8 metastable states.

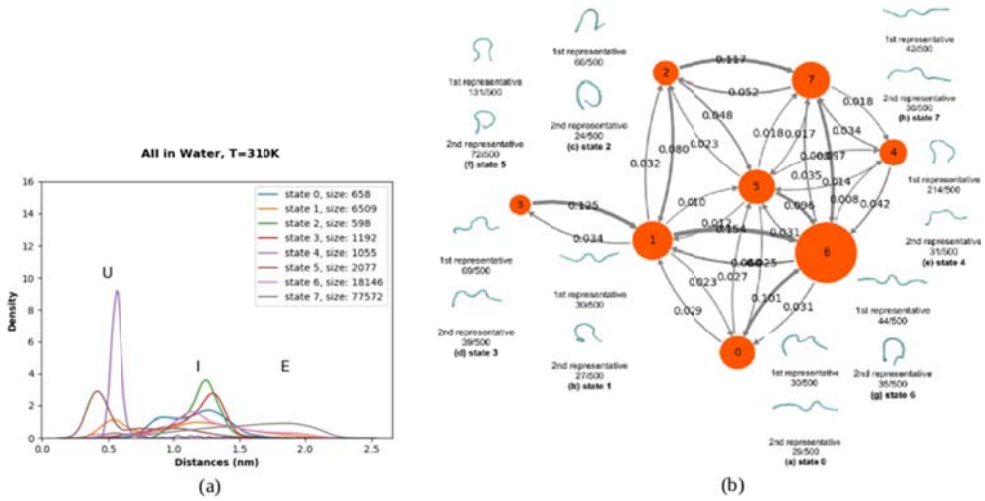


Figure 5. MSM for All in Water at 310K. (a) Probability Density plots of end-to-end distances for each macrostate. (b) Kinetic model of MSM with representatives for each state using GROMOS.

Comparing the probability densities of the resulting macrostates from the Markov State Models, we observe that in the Water/Ethanol solvent there is a preference towards more compact U-shaped metastable states.

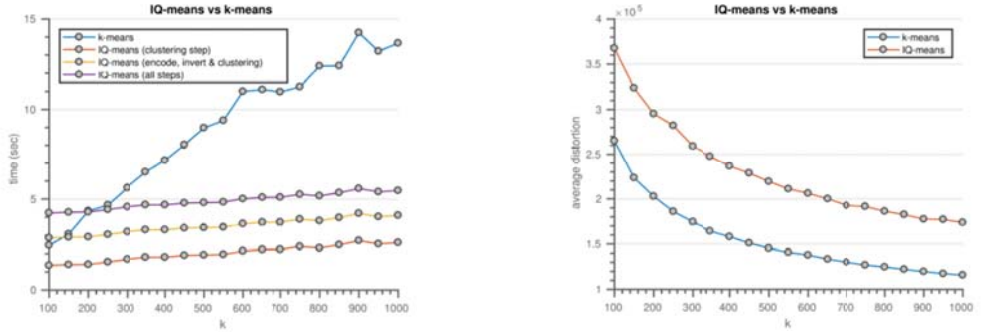
3.2.2 MSM for AII using IQ-means for the generation of the microstates

The microstates here are generated by IQ-means using the same features and TICA components that were clustered previously with k-means. We compare the different clusterings and validate the microstates from the approximate method by comparing the resulting macrostates.

k-means vs IQ-means on TICA coordinates

IQ-means is compared against k-means in terms of time and quality (Fig. 6). The quality comparison of IQ-means clusters is evaluated using the cost function of k-means as a distortion measure (Fig. 6b). IQ-means is set at 20 iterations, as suggested in [1] and the codebook was learned with a random sample of 5000 points from the dataset. Time and distortion was measured for 100 to 1000 clusters with a step of 100. Every experiment was performed 3 times for reliability and the mean values are reported. For IQ-means there are 3 plots in Fig. 6a, since some preprocessing steps are not needed at every run. Only, the clustering step is always necessary.

The time gain is orders of magnitude greater for many clusters, with fair loss in distortion (1.5 times worse than k-means), making it a reasonable trade-off.



(a) k (number of clusters) vs. time (sec)

(b) average distortion vs. k (number of clusters)

Figure 6. IQ-means vs k-means.

MSM with IQ-means

Afterwards, we build the MSM with the 100 microstates produced by IQ-means and compare the final metastable states, since similar discretizations should result to equal MSMs. Our results (Fig. 7, 8) show that the probability densities for MSM with k-means and IQ-means for AII in both solvents at 310K appear to be alike (Fig. 7a,c, Fig. 8a,c), especially for U-shaped meta-states.

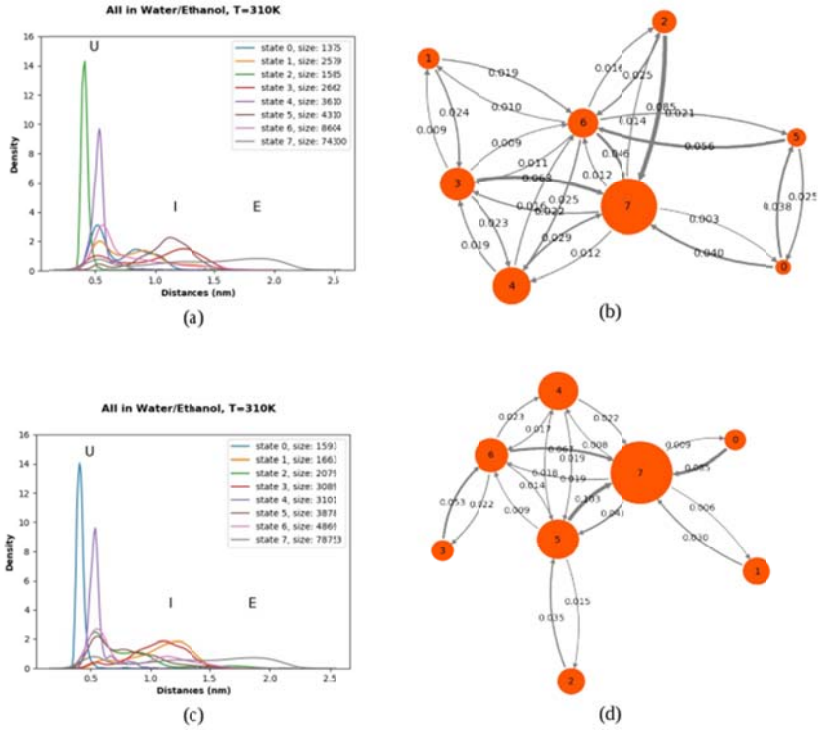


Figure 7. MSM for All in Water/Ethanol, T=310K using k-means with 100 clusters (a,b) and IQ-means with 100 clusters (c,d).

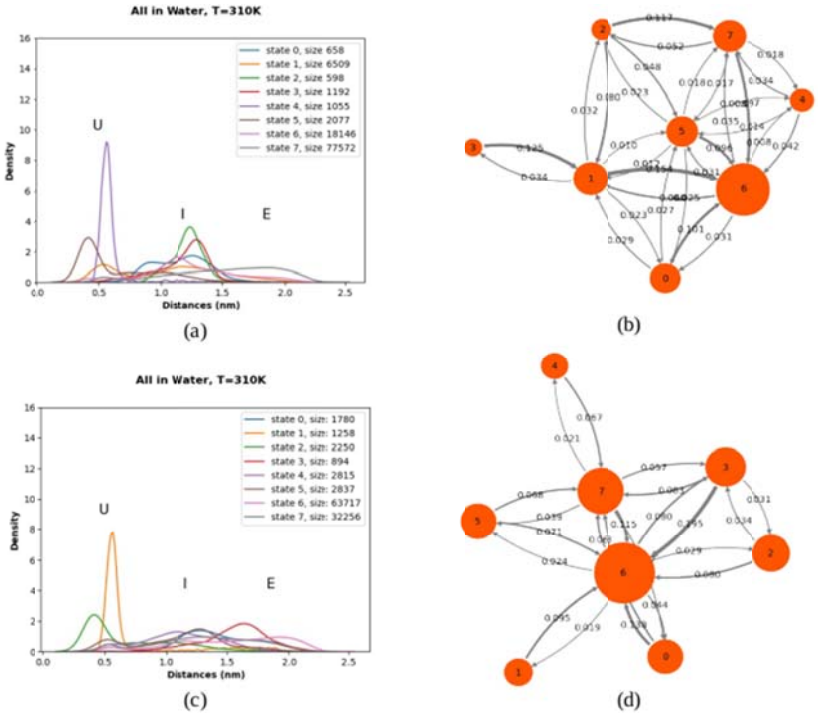


Figure 8. MSM for All in Water, T=310K using k-means with 100 clusters (a,b) and IQ-means with 100 clusters (c,d).

3.3 Metadata representatives

We apply MDS on the matrix of the backbone atoms distances (Fig. 9a) for each conformation of AII in Water/Ethanol at 310K. Coloring each 3D point by the corresponding End-to-End distance, indicates that these representations preserve the various structures that exist in the simulation (Fig. 9b). The darker red points (smaller end-to-end distances) are the U-shaped conformations, while the lighter yellow points (bigger end-to-end distances) are the Extended (E). Intermediate colors correspond to the I-shaped conformations. Also, in the figure we observe the results of k-means and IQ-means clustering on these representations (Fig. 9c,d).

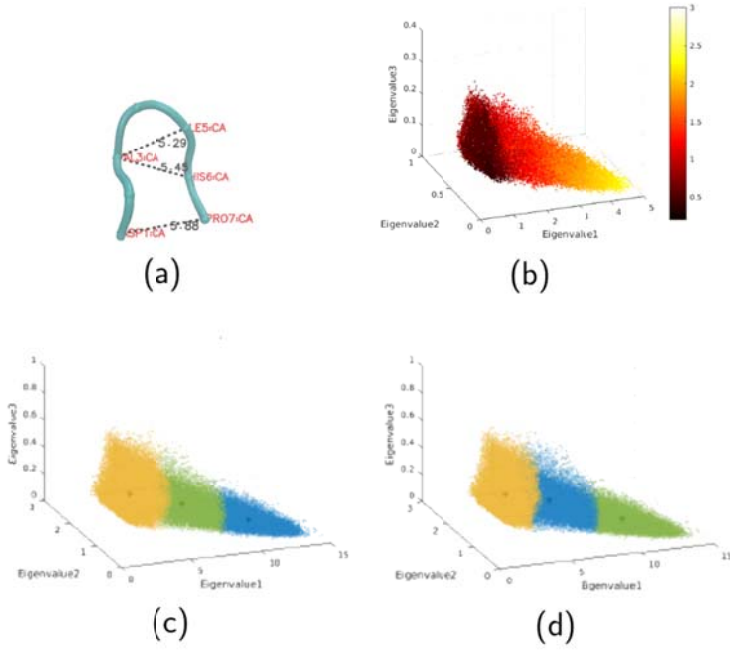


Figure 9. Metadata representatives. (a) Subset of the backbone atoms distances used for MDS. (b) 3D points metadata representatives colored by End-to-End distances. (c,d) Another comparison of k-means (c) and IQ-means clustering (d).

4. CONCLUSIONS

We investigated the bioactive conformation of the human Angiotensin II and how it is achieved, using Molecular Dynamics simulations in different solvents (Water and Water/Ethanol) and temperatures (278K, 298K, 310K, 323K). Our results show that in the membrane environment the peptide tends to a compact U-shaped structure, specially for T=310K, while in Water there were smaller populations of folded U-shaped structures.

The eigenvalues from the Multidimensional Scaling of the backbone atom distances, appear to produce efficient metadata representations for each conformation as a single 3D point. In the plots for Angiotensin II, the representations appear to distinguish the folded U-shaped structures from the intermediate and extended structures.

Also, we validated the efficiency, in time and quality, of an inverted-quantized k-means approximation algorithm (IQ-means) using the conformations from the MD simulations. IQ-means seems to provide a very fast approximation for k-means with a fair loss in quality.

To identify the meta-stable states we constructed Markov State Models for Angiotensin II in Water and Water/Ethanol at T=310K. The results indicate that in Water/Ethanol there are more metastable states represented by U-shaped structures than in the Water solvent. Also, the macrostate representatives generated by GROMOS clustering appear to give a good intuition for the amount and the variance of the structures in each meta-stable state.

Finally, the microstates generated by IQ-means seemed to produce similar MSMs with k-means' microstates. Therefore, we are able to approximate the MSMs using IQ-means with a fair trade-off between time and accuracy, especially for very long trajectories (big data). Furthermore, IQ-means could be used to quickly identify the number of microstates in a trajectory, before generating them with more time-consuming methods, such as k-means.

The results of this work have been presented at the poster session of BioExcel's 2nd SIG Meeting "Advanced Simulations for Biomolecular Research", a satellite event of the European Conference on Computational Biology (ECCB) 2018 in Athens, Greece.

REFERENCES

[1] Y. Avrithis, Y. Kalantidis, E. Anagnostopoulos, and I.Z. Emiris. Web-scale image clustering revisited. In Proceedings of International Conference on Computer Vision (ICCV 2015), Santiago, Chile, December 2015.

[2] M.K. Scherer, B. Trendelkamp-Schroer, et al., F. Noé: PyEMMA 2: a software package for estimation, validation, and analysis of Markov models J.Chem. Theory Comput., 11 (2015), pp. 5525-5542

[3] Spyroulias, G. A.; Nikolakopoulou, P.; Tzakos, A.; Gerothanassis, I. P.; Magafa, V.; Manessi-Zoupa, E.; Cordopatis, P., Comparison of the solution structures of angiotensin I & II. Implication for structure-function relationship. European journal of biochemistry / FEBS 2003, 270 (10), 2163-73.

[4] Daura X, Gademann K, Jaun B, Seebach D, van Gunsteren WF, Mark AE. Peptide Folding: When Simulation Meets Experiment. Angew Chem Int Ed. 1999;38:236–240. doi: 10.1002/(SICI)1521-3773(19990115)38:1/2<236::AID-ANIE236>3.0.CO;2-M

[5] Schütte, C., Fischer, A., Huisinga, W. and Deuflhard, P.: A Direct Approach to Conformational Dynamics based on Hybrid Monte Carlo. J. Comput. Phys. 151, 146-168 (1999)

[6] Bowman GR (2012) Improved coarse-graining of Markov state models via explicit consideration of statistical uncertainty. J Chem Phys 137:134111

[7] S. Röblitz and M. Weber: Fuzzy spectral clustering by PCCA+: application to Markov state models and data classification. Adv. Data Anal. Classif. 7, 147-179 (2013).

[8] P. Deuflhard and M. Weber: Robust Perron cluster analysis in conformation dynamics.. In: Linear Algebra Appl.. M. Dellnitz, S. Kirkland, M. Neumann and C. Schütte (editors). Elsevier, New York, 2005. 398C, (2005).

[9] Noé , F.; Wu, H.; Prinz, J.-H.; Plattner, N. Projected and Hidden Markov Models for calculating kinetics and metastable states of complex molecules. J. Chem. Phys. 2013, 139, 184114

[10] Boyu Zhang, Trilce Estrada, Pietro Cicotti, and Michela Taufer. 2014. Enabling In-Situ Data Analysis for Large Protein-Folding Trajectory Datasets. In Proceedings of the 2014

IEEE 28th International Parallel and Distributed Processing Symposium (IPDPS '14). IEEE Computer Society, Washington, DC, USA, 221-230.
[11] Prinz, J.-H.; Keller, B.; No, F. Probing molecular kinetics with Markov models: metastable states, transition pathways and spectroscopic observables. Phys. Chem. Chem. Phys. 2011, 13, 16912–16927.
[12] R Bowman, G.; Beauchamp, K.; Boxer, G.; Pande, V. Progress and challenges in the automated construction of Markov state models for full protein systems. The Journal of chemical physics 2009, 131, 124101.

Contributing to the pathway towards 5G experimentation with an SDN-controlled network box

Dimitrios G. Dimopoulos (ddimopoulos@di.uoa.gr, jim.dimopou@gmail.com)

ABSTRACT

This diploma thesis aims at presenting the “Network in a box”, an innovative tool we developed which is based on the key 5G principles, SDN and NFV. With Software Defined Networking (SDN) being the new approach in mobile networks, control and data plane are decoupled providing the ability to make any control related decisions centrally and transform legacy network devices to simple forwarding elements. This testbed is a portable emulated network device which is self-managed and self-optimised and can be connected between any real network devices, emulating how the 5G network will perform. This plug & play black-box testbed is also capable of providing KPI metrics of the 5G network under real circumstances when real network devices are connected to it.

Keywords: 5G, Software Defined Networks, Network Functions Virtualisation, MANO, Network in a box, Openflow, Mininet, POX

Advisors
Lazaros Merakos, Professor (University of Athens)

1. INTRODUCTION

The future of mobile communications is likely to be very different to what we are used to today. While demand for mobile broadband will continue to increase, we are already seeing the growing impact of technology as things around us become ever more connected. Next generation networks will have to integrate networking, computing and storage resources into one programmable and unified infrastructure. This unification will allow for an optimized and more dynamic usage of all distributed resources as well as the convergence of fixed, mobile and broadcast services. An adaptive network solution framework will become a necessity for accommodating both LTE and air interface evolution;

Cloud, SDN and NFV technologies will reshape the entire mobile ecosystem and speed up the creation of massive-scale services and applications.

Conventional networks are characterized by a static architecture not able to deal with the dynamic and always changing needs. Populated with a large and increasing variety of proprietary hardware appliances, traditional networks often require numerous devices in order to launch a new network service, thus increasing the overall network complexity. To this end, an alternative networking approach is essential to manage complex multi-layer and multi-technology networks and achieve built-in flexibility.

2. PROPOSED APPROACH

Software Defined Networking (SDN) will play a vital role in future mobile networks, towards addressing all those challenges imposed by conventional networks' limitations. Traditionally the control plane of a network, which is responsible for managing the routing and flow of data, was implemented at a hardware level. As a result, altering the behaviour of a network, required reconfiguration of a vast number of devices each containing vendor specific protocols; a costly process in terms of both time and money. SDN decouples the control plane from the data plane (the actual network traffic), allowing centralised control over the behaviour of the entire network.

The rules for handling data can now be specified in software at the controller, which communicates with the data plane (i.e., switches, routers) through an open interface. As a result, it is possible to alter the entire behaviour of the network from a single logical point without needing to physically touch the hardware. This allows for greater efficiency in the utilisation of resources as the network can be reprogrammed to meet current demands. SDN is a key component of the 5G vision of flexible networks and will have profound implications on the manner according to which resources are allocated and managed.

The essence of SDN is possibly best characterised by four of its core principles:

- *Decoupling of control and data planes*
  This principle is the foundation of the SDN concept. It advocates the separation of the control plane into a logically centralised software controller which is capable of managing and altering the routing of data through the network. This separation has an implicit implication that the controller is in some way external to the physical equipment that it controls. Decoupled data and control planes co-located on the same device blurs the definition of SDN.
- *Logically centralised controller*

The extracted control plane is logically centralised into a single controller with a network wide view.

This logically centralised controller may in fact consist of multiple virtual or physical controllers operating in a distributed manner, depending on the scale of the network.

- *Open interfaces*
  One of the motivating factors behind SDN was to reduce the effort and cost associated with reconfiguring the vendor-specific devices in the network. An open, standardized interface between devices in the control and data planes, known as the southbound application program interface (API), is therefore a key principle of SDN.
- *Programmability by external applications*
  The controller in SDN allows for programmability by external applications through the so-called northbound API. This naturally lends itself to the concept of adaptability. It allows the network operator to view the myriads of physical hardware under its control as a single programmable entity which it can configure.

A more comprehensive overview of SDN and its implications in terms of programmable networks is provided through a comparison between two of the most popular SDN protocols; OpenFlow and ForCES.

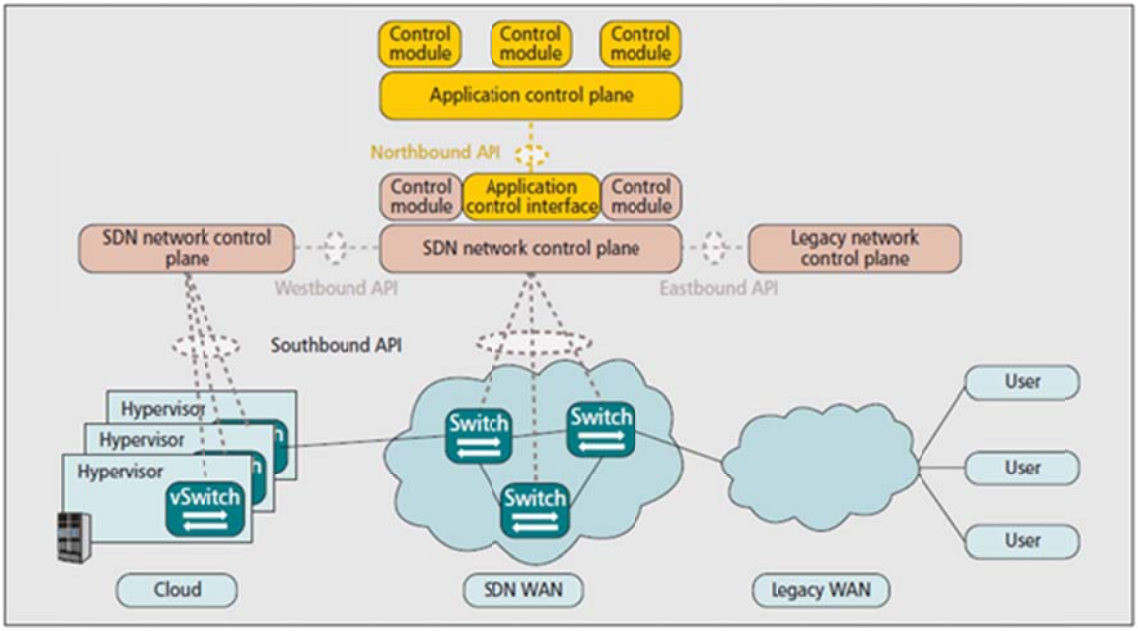


Figure 1: The high-level SDN architecture

3. NETWORK FUNCTIONS VIRTUALISATION (NFV)

Network Functions Virtualisation (NFV) refers to the implementation of network functions in software running on non-proprietary, commoditized hardware. This approach allows the deployment of network functions in data centres and leverages IT virtualisation technology to separate network functions from the underlying hardware. In other words, previously discrete and vertically integrated network elements can be implemented in a cloud platform to form a private Infrastructure as a Service (IaaS).

Implementing network functions in software on general purpose computing/storage platforms will allow for new flexibilities in operating and managing mobile networks. In mobile networks, NFV is currently discussed in the context of virtualising the core network. Furthermore, NFV and implementing mobile network functions in data centres allows more flexibility in terms of resource management, assignment, and scaling. This has also an impact on the energy efficiency of networks as only the required amount of resources may be used and overprovisioning of resources can be avoided. This resource orchestration could reuse management algorithms already developed in the IT world in order to exploit resources as efficiently as possible.

As mentioned, NFV is already applied on core networks and first trials are performed demonstrating that critical mobile network functions such as MME, PGW, or HSS can be implemented on standard IT platforms. A critical enabler of this development is, besides virtualisation technologies, the availability of highspeed IP networks and the possibility to manage them more flexibly through SDN.

4. SELF-ORGANISING NETWORKS

To achieve built-in flexibility current networks will transform from comprising vertically integrated discrete network elements, to being cognitive, cloud optimised and seamless in operation. Future networks should be able to optimise themselves autonomously. Self-organising capabilities enable the network to efficiently predict demand and to provide resources, so that it can heal, protect, configure and optimise accordingly. The platforms will do this by generating the minimum cost on network equipment (CAPEX) and operations cost (OPEX), whilst keeping QoS tailored to user demand with adequate resources. The overall objective is to create a cognitive and autonomic management system developed through the application of policies that can self-adapt to the changing conditions of the network and to the external environment in which the network operates, via a well-defined set of self-organising

functions. These platforms also need to support multi-tenancy environments. Self Organising Networks (SON) are the

are necessary in order to meet all those strict requirements imposed by 5G, are now available through the benefits originated from the NFV and SDN penetration. With NFV being the key enabler to develop software-based network functions on commodity hardware by leveraging virtualisation techniques and SDN bringing the ability to decouple the control from the data plane through the usage of a logically centralised controller, 5G Architecture needs can become reality.

To this direction, we developed an experimental testbed in order to emulate the 5G network capabilities using state-of-the-art techniques so as to deploy and retrieve KPI metrics from what is expected to be the norm in next generation mobile networks. Our testbed is comprised of a portable server which emulates 5G network by deploying a custom topology of OVS's, as complex as the user decides to be and can be used as plug & play machine in real customer environments where it demonstrates how customer network will behave under real conditions such as continuously heavy or burst traffic, link degradations or failures and even more.

Leveraging from the existing Mininet emulator capabilities and the python-based POX controller, our portable testbed is qualified to go beyond and reveal the real 5G network demands and capabilities either if it is first step towards the automation of networks' operation, administration and maintenance tasks, introducing closed control loop functions dedicated to self-configuration, self-optimisation, and self-healing. Towards this direction we developed an innovative solution which is thoroughly presented in the next section.

5. SDN – NETWORK BOX

As it was thoroughly analysed in the previous chapters, 5G demands in terms of latency, capacity, coverage and heterogeneity are extreme. The technology advances in network architecture design which anticipated to be used in order to emulate the 5G Access, Transport or Core Network.

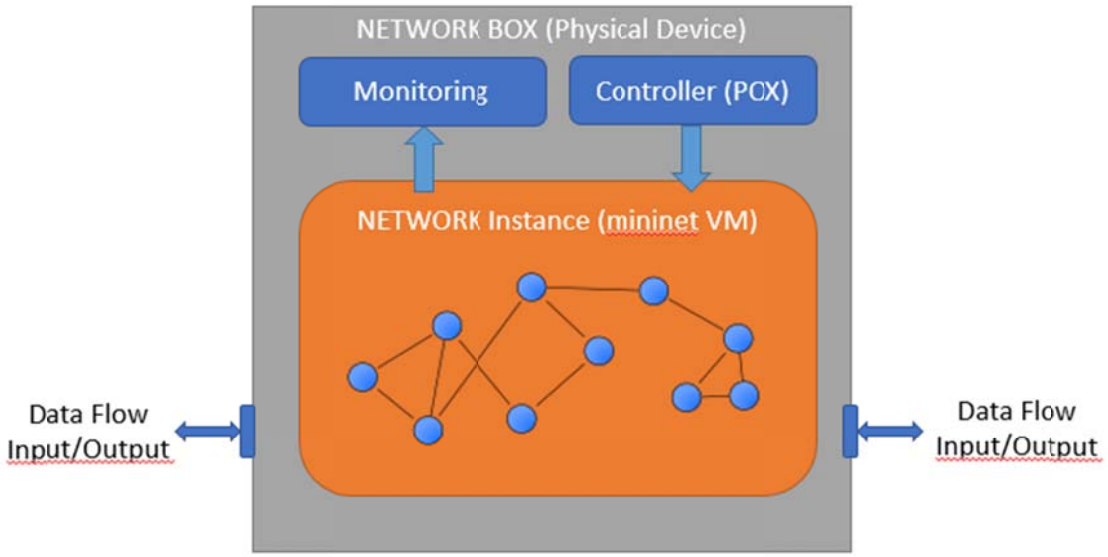


Figure 2: The SDN-based Network BOX

As it is already well-known, Mininet emulator can be used to create a custom network topology of switches and internal hosts, where hosts can establish communication and are able to exchange data in a way that emulates end-to-end network topology. This approach is great but not as realistic as it happens to be in real networks. In a real scenario, the network topology will be somehow complex with possibly hundreds of switches being interconnected and multiple links between them, and hosts are always real elements, i.e. standalone real servers or virtual machines which may not necessarily be part of the network topology as Mininet currently requires them to be. Mininet emulator provides the ability to configure the network topology (number of switches and links between them), in a custom way but also requires hosts' configuration and connection to switches in order to have a functional emulated network.

But what if the hosts are not part of the topology as it always happens to be when we are talking about real networks and particularly 5G? Here comes our innovative solution which goes beyond the existing mininet capabilities and by leveraging 5G main pillars, SDN & NFV state-of-the-art techniques, provides an implementation where two real hosts, for example two separate Linux-based VMs, are interconnected and exchange data through our portable testbed. In this way we succeed in having a real black-box network which can be interleaved anywhere into customer network infrastructure, providing the ability to connect multiple hosts to our testbed's external interfaces, administered by different NICs, and establish communication.

This last chapter presents in detail our innovative implemented idea, the so-called **"Network in a box"**. Our approach provides the ability to create a custom network topology, as complex as we decide it to be, based on Mininet VM image which is connected and managed by POX controller (remote controller running on Mininet VM), in a portable server configured in a custom way, where we could connect two hosts and check how they communicate with each other through a network which always adapts to possible changes such as link failures/delays, QoS type etc., exploiting SDN capabilities.

Network in a box proof of concept

In our solution we use a single server (Host) with the following characteristics:

- Ubuntu 16.04 LTS
- Memory: 16GB
- Processor: Intel Core i5-6500 CPU @ 3.20GHz x 4
- Graphics: Intel HD Graphics 530 (Skylake GT2)
- OS type: 64-bit
- Disk: 250 GB SSD

For hardware abstraction purposes, the virtualisation application we use is Virtual Box released by Oracle (Version 5.0.40_Ubuntu r115130)

Our solution includes 3 different VMs:
- 2 Ubuntu 16.04 64-bit VMs with 2048 MB of RAM, 30GB disk and 1vCPU and
- the Mininet VM: 3GB RAM, 8GB disk

Network Configuration

In all 3 VMs we have configured two Network Adapters in the following way:

1st Network Adapter in bridged mode attached to external physical host's interface

2nd Network Adapter configured as Host-only used for internal communication between VMs in VBox and for communication of guest VMs with the host (Ubuntu server which is our testbed).

As far as the testbed is concerned, we have explicitly configured it with the aforementioned Mininet VM details with regards to Memory and Disk size.

Using VBox we deploy two guest VMs (VM1 and VM2) which are real external hosts in our experiment and the mininet-based testbed being interleaved between them, so traffic generated between the hosts will be directed through the testbed network. We start all three VMs from VBox: VM1, VM2 and testbed server with the configuration provided in the screenshots above. When each VM

boots up, we check the VBox internal interface IP address and connect to that VM from our Ubuntu Host (server machine).

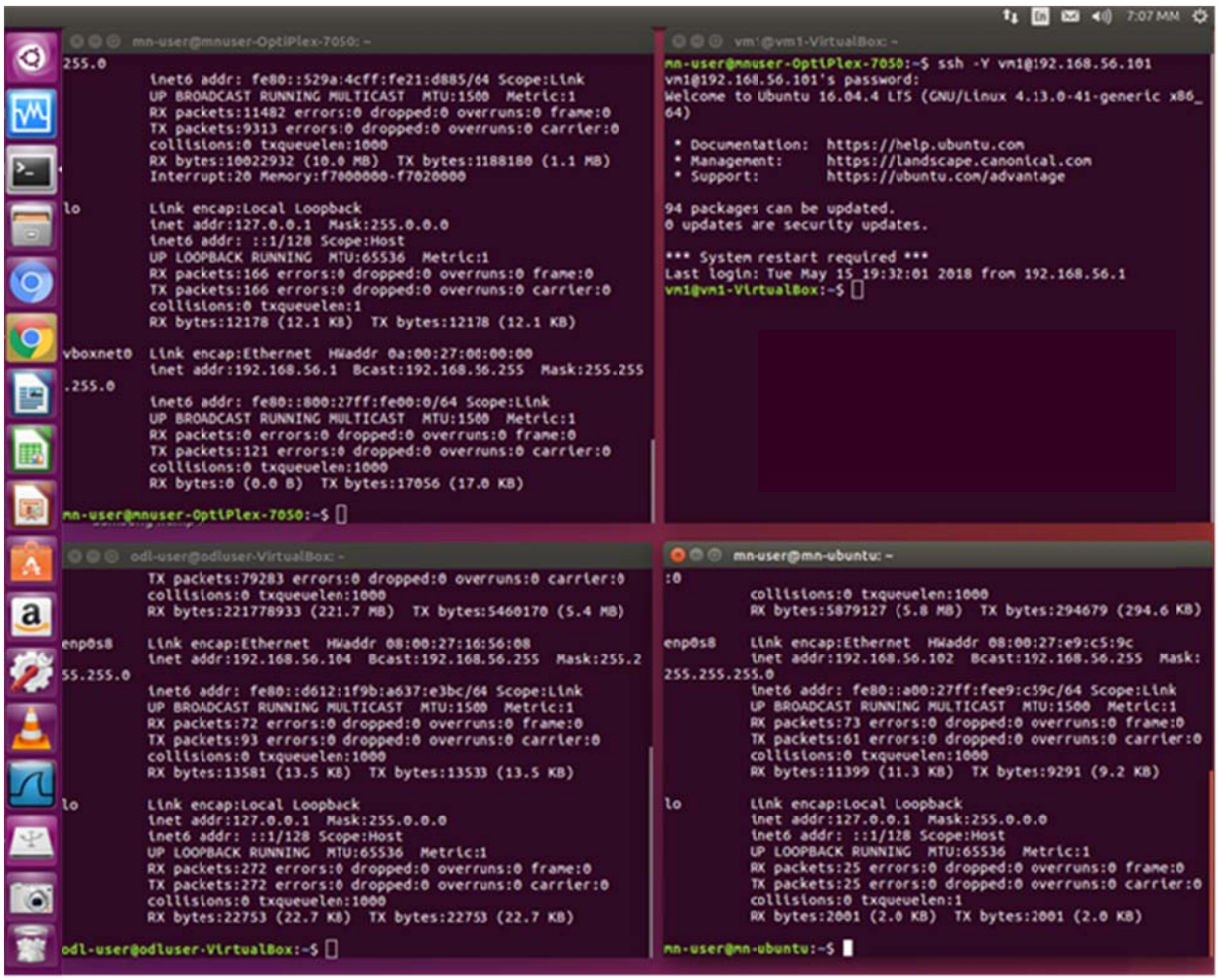


Figure 3: Depicting connections to testbed VM, guest VMs and host machine

For our testbed VM we will open 3 terminal windows in total (one for mininet cli, one for POX remote controller and one for flow entries' configuration of OVS's)

POX installation

In our testbed VM running mininet image, we install Pox controller and then we start it, issuing the appropriate command in one of the mininet-VM terminals.

The final solution we follow in order to achieve the expected behaviour, is to bring up a mininet topology with 6 switches and 2 internal mininet hosts using our custom python script named *custom_heavy5_br.py* which has been created under mininet/examples directory in mininet-vm.

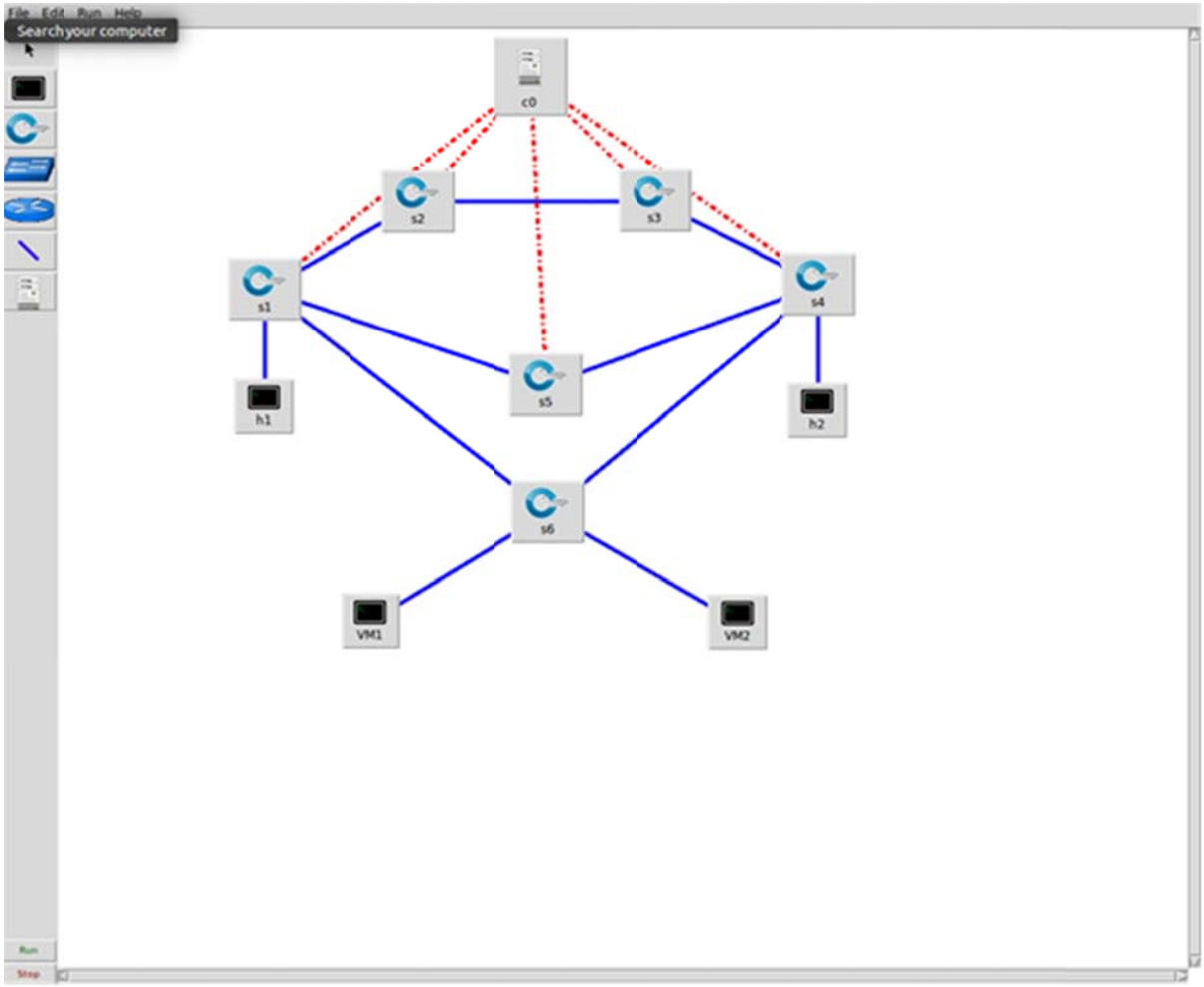


Figure 4: ***Network in a box*** **proposed topology**

In the portable testbed VM, Switch s6 acts as bridge where the two Host VMs (VM1 and VM2) will be connected to. At this point we would like to mention that internal mininet hosts h1 and h2 are deployed for mininet topology bring-up purposes and will not be used at all throughout our experiment.

Once mininet topology is brought up, we are going to see Openflow messages being exchanged between mininet OVS's and POX remote controller. Connection of controller and mininet switches has been properly established.

Going forward, we will now apply ovs flow entries for each OVS, so as to achieve two data paths; one main path through s1→s2→s3→s4 and a failover path which is going to be used in case of a link failure or degradation below a threshold, s1→s5→s4.

In this way, assuming that VM1 sends some traffic to VM2, the data path to be followed will be:
VM1→s6→s1→s2→s3→s4→s6→VM2, (if main path is going to be used),
or
VM1→s6→s1→s5→s4→s6→VM2, (in case failover path is going to be followed)

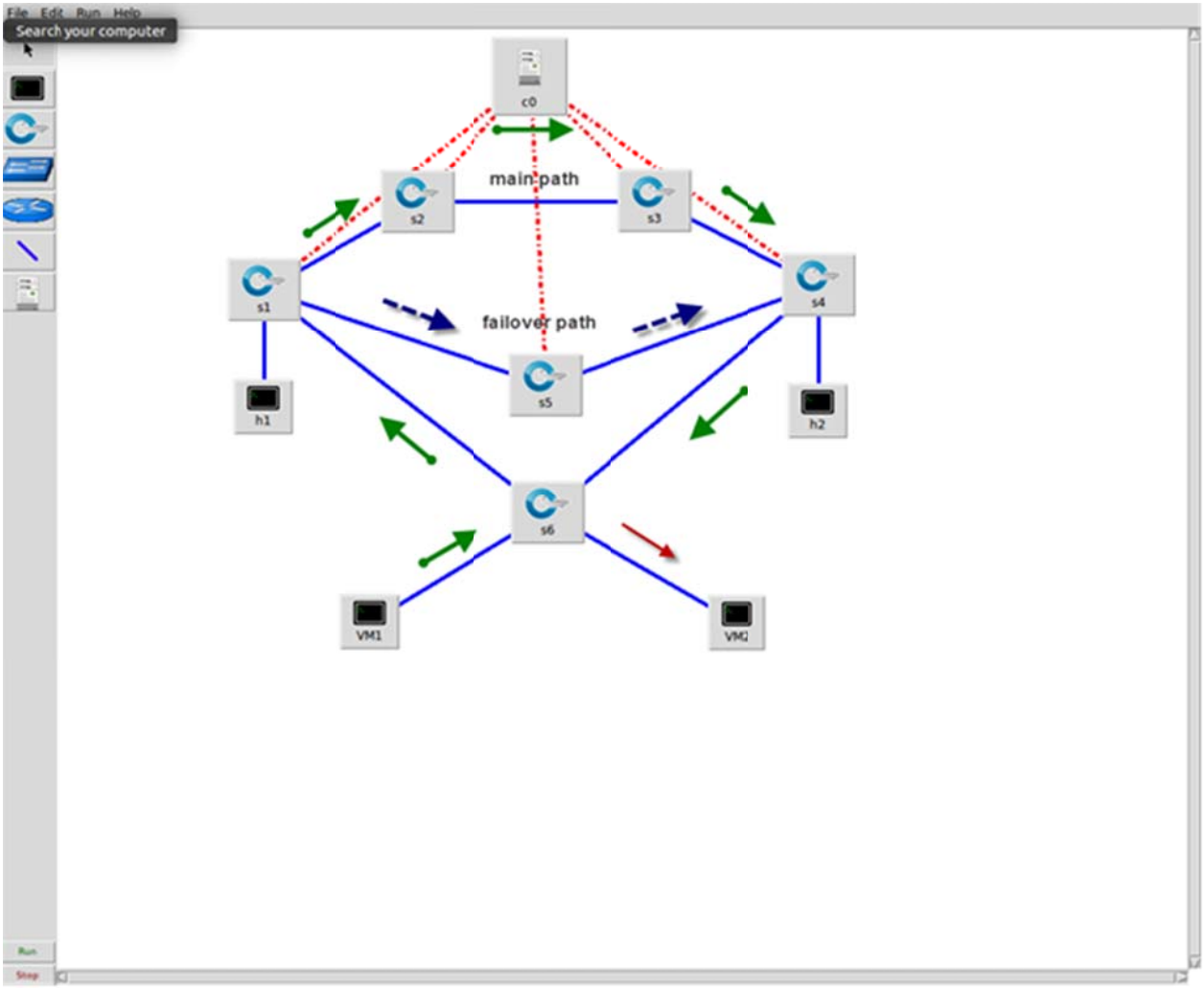


Figure 5: OVS Topology and possible routes from VM1 to VM2

Experiments

In our first experiment VM1 is the server and VM2 acts as client.
We use iperf command in both VMs to establish and verify connectivity of the VMs while traffic is passing by mininet OVS's. We open a terminal, connect to mininet-VM and then start pox controller.
POX controller will be successfully started and listening to port 6633.

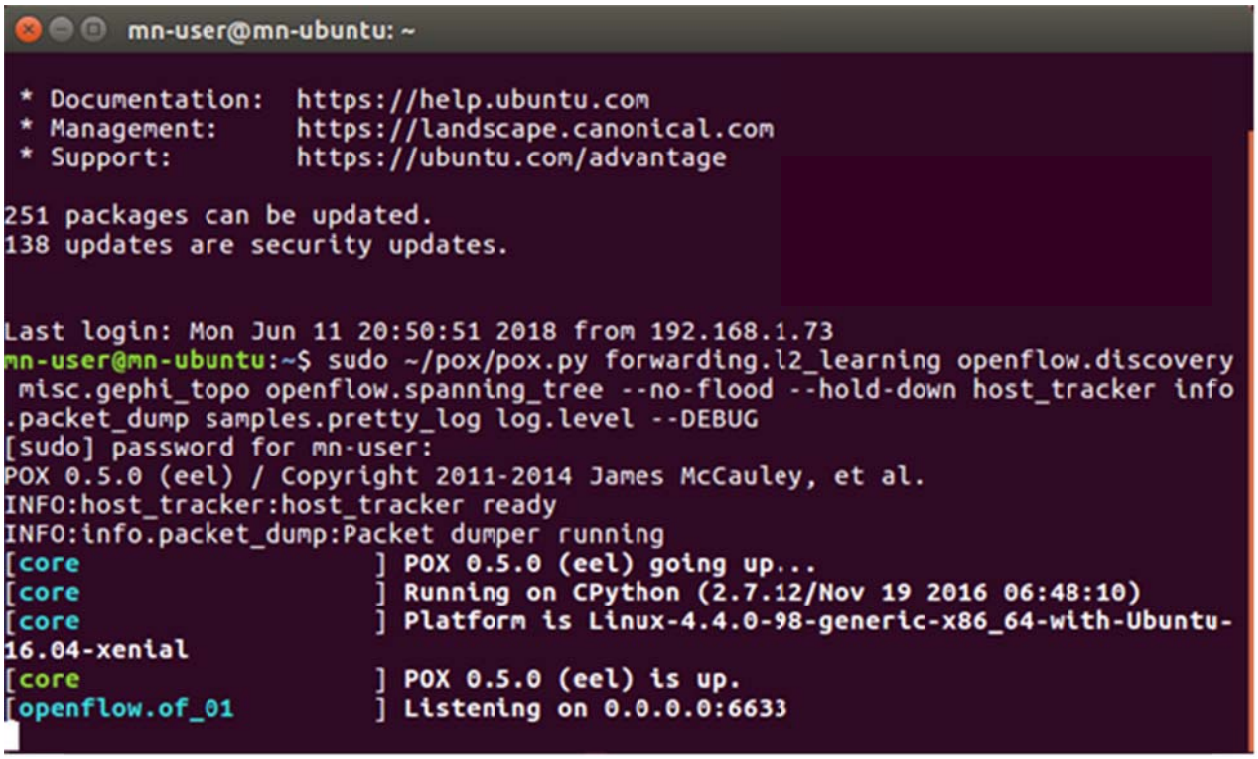


Figure 6: POX controller startup

Then we start custom mininet topology in our testbed and connect it to pox controller which is also running on the same VM (testbed VM). Mininet topology is brought up and at the same time POX controller starts displaying Openflow messages between controller and virtual switches (OVS's)

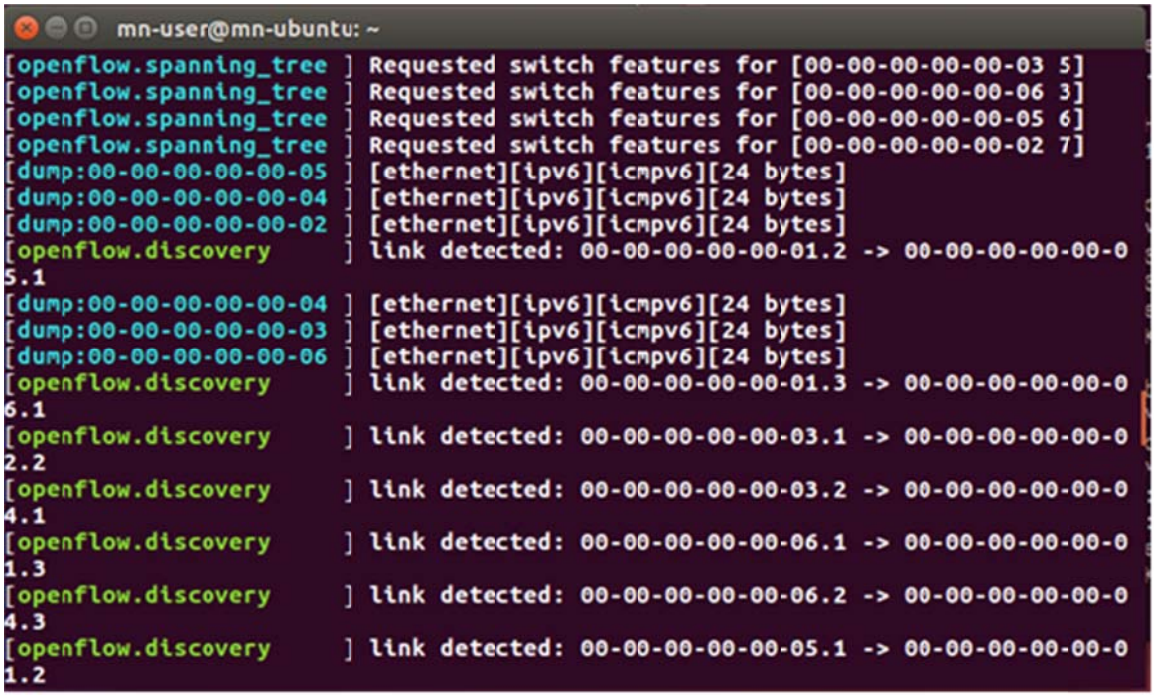


Figure 7: POX flow messages and link discovery

Mininet topology contains 6 OVS, where s6 acts as main bridge.
We can now check mininet topology using the following commands in mininet prompt: *nodes*, *links*, *dump*

```
mininet> nodes
available nodes are:
c0 h1 h2 s1 s2 s3 s4 s5 s6
mininet>
```

Figure 8: Mininet nodes

```
mininet> links
h1-eth0<->s1-eth4 (OK OK)
h2-eth0<->s4-eth4 (OK OK)
s1-eth1<->s2-eth1 (OK OK)
s1-eth2<->s5-eth1 (OK OK)
s1-eth3<->s6-eth1 (OK OK)
s2-eth2<->s3-eth1 (OK OK)
s3-eth2<->s4-eth1 (OK OK)
s4-eth3<->s6-eth2 (OK OK)
s5-eth2<->s4-eth2 (OK OK)
mininet>
```

Figure 9: Mininet links

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=2148>
<Host h2: h2-eth0:10.0.0.2 pid=2151>
<OVSSwitch{'protocols': 'OpenFlow10'} s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None,s1-eth4:None pid=2157>
<OVSSwitch{'protocols': 'OpenFlow10'} s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None pid=2160>
<OVSSwitch{'protocols': 'OpenFlow10'} s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None pid=2163>
<OVSSwitch{'protocols': 'OpenFlow10'} s4: lo:127.0.0.1,s4-eth1:None,s4-eth2:None,s4-eth3:None,s4-eth4:None pid=2166>
<OVSSwitch{'protocols': 'OpenFlow10'} s5: lo:127.0.0.1,s5-eth1:None,s5-eth2:None pid=2169>
<OVSSwitch{'protocols': 'OpenFlow10'} s6: lo:127.0.0.1,s6-eth1:None,s6-eth2:None pid=2172>
<RemoteController{'ip': '127.0.0.1', 'port': 6633} c0: 127.0.0.1:6633 pid=2142>
mininet>
```

Figure 10: Mininet dump

Open s6 with xterm and execute: ./connect.sh (script details in ANEX I) Script is used to bind s6 switch to external physical interface enp0s3, acting as bridge. A new window will come up where we execute the connect.sh script.

The script should be customized accordingly based on the Host's external interface allocated IP address and the gateway IP. Then, flow entries' configuration follows which enables traffic routing through the main or failover path with higher priority selected on the main path. Detailed OVS flow table/entry configuration based on Openflow protocol specifications, is provided in ANEX II. Once flow entries are configured in all OVS's, we are ready to start

the test. As mentioned earlier, VM1 acts as the iperf client and VM2 acts as the iperf server.

```
vm1@vm1-VirtualBox:~$ iperf -c 192.168.56.104
------------------------------------------------------------
Client connecting to 192.168.56.104, TCP port 5001
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 192.168.56.101 port 33848 connected with 192.168.56.1
04 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  2.02 GBytes  1.74 Gbits/sec
vm1@vm1-VirtualBox:~$
```

Figure 11: VM1 - iperf client

```
odl-user@odluser-VirtualBox:~$ iperf -s
------------------------------------------------------------
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
------------------------------------------------------------
iperf -c 192.168.56.104iperf -c 192.168.56.104iperf -c 192.168.56.
104[  4] local 192.168.56.104 port 5001 connected with 192.168.56.
101 port 33848
[ ID] Interval       Transfer     Bandwidth
[  4]  0.0-10.0 sec  2.02 GBytes  1.74 Gbits/sec
```

Figure 12: VM2 - iperf server

6. CONCLUSION

A very important factor in the evaluation of SDN is the customer perception of the 5G networking technology which is based on the capabilities exposed and KPI metrics collected under specific use cases, which in turn provide the ability to quantify user perception of the network capabilities through QoS and QoE.

To this end, we introduced an innovative idea developed in our university's communications network research laboratory, an autonomous emulated portable network testbed, the SDN based **Network in a box.** It is a portable plug & play testbed device capable to be interconnected to any legacy network component and emulate how 5G network performs under real circumstances, providing also the ability to extract KPI metrics from the examined topology. It presents how the 5G network behaves upon link degradation or link failures while traffic keeps being managed by the self-organised network capabilities without interruption. The implemented testbed device has been presented and

evaluated and its contribution goes far beyond an abstract framework introduction, as it provides a practical implementation of real-time SDN based 5G network which apart from its already notable capabilities, can be used as a proof of concept for application related experiments in the testbed's northbound interface and provide application related metrics under various scenarios which can take place in the emulated network.

REFERENCES

[1] H. Koumaras et al., "Enabling Agile Video Transcoding over SDN/NFV-enabled Networks", International Conference on Telecommunications and Multimedia (TEMU), July 2016.
[2] T. Yu et al., "Adaptive Routing for Video Streaming with QoS Support over SDN Networks", International Conference on Information Networking (ICOIN), January 2015.
[3] L. Sørensen, K. Skouby, "Visions and research directions for the Wireless World", July 2009, pp. 5-9.
[4] C. Wang et al., "Cellular Architecture and Key Technologies for 5G Wireless Communication Networks", IEEE Communications Magazine, February 2014.
[5] Σ. Γ. Μαστοράκης, "Μέθοδοι εξουσιοδότησης για δέσμευση πόρων σε Ευφυή – Προγραμματιζόμενα - Δίκτυα (Software-Defined-Networks)", May 2014.
[6] T. Zinner et al., "Dynamic application-aware resource management using Software-Defined Networking: Implementation prospects and challenges", IEEE Network Operations and Management Symposium (NOMS), May 2014.
[7] J. Andrews et al., "What Will 5G Be?", IEEE JSAC SPECIAL ISSUE ON 5G WIRELESS COMMUNICATION SYSTEMS, May 2014.
[8] Open Networking Foundation (ONF), "Software-Defined Networking (SDN) Definition", https://www.opennetworking.org/sdn-resources/sdn-definition, 2017.
[9] Open Networking Foundation (ONF), "SDN Architecture", Issue 1.0
[10] "ONF OpenFlow Switch Specification v1.5.1"
[11] "Using the POX SDN Controller", http://www.brianlinkletter.com/using-the-pox-sdn-controller/
[12] "POX Controller Tutorial", http://sdnhub.org/tutorials/pox/
[13] "Download/Get Started with Mininet", http://mininet.org/download/, 2017.
[14] "Set up the Mininet network simulator", http://www.brianlinkletter.com/set-up-mininet/
[15] "A Mininet-based Virtual Testbed for Distributed SDN Development, SIGCOMM 2015", https://conferences.sigcomm.org/sigcomm/2015/pdf/papers/p365.pdf
[16] "A Network in a Laptop: Rapid Prototyping for Software-Defined Networks, SIGCOMM 2010", https://conferences.sigcomm.org/hotnets/2010/papers/a19-lantz.pdf
[17] "Reproducible Network Research with high-fidelity emulation, Stanford", https://stacks.stanford.edu/file/druid:zk853sv3422/heller_thesis-augmented.pdf
[18] Oracle VirtualBox
[19] "Installing new version of Open vSwitch", https://github.com/mininet/mininet/wiki/Installing-new-version-of-Open-vSwitch
[20] I. Mustafa and T. Nadeem, "Dynamic Traffic Shaping Technique for HTTP Adaptive Video Streaming using Software Defined Networks", IEEE International Conference on Sensing, Communication and Networking (SECON), June 2015.
[21] "Dynamic Traffic Prioritization in IoT networks using SDN (ONOS Controller and Mininet Topology) – Github", https://github.com/Y0Username/iotDynamicPri
[22] "Openvswitch and ovsdb", https://sreeninet.wordpress.com/2014/01/02/openvswitch-and-ovsdb/
[23] "Administer OpenFlow Switches - Ubuntu", http://manpages.ubuntu.com/manpages/xenial/en/man8/ovs-ofctl.8.html
[24] "Openflow for life - Github", https://github.com/kevinvkell/openflow-for-life

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Εθνικόν και Καποδιστριακόν Πανεπιστήμιον Αθηνών,
Πανεπιστημιούπολη, 15784 Αθήνα