Structure and Dynamics of Type III Secretion Effector Protein ExoU as determined by SDSL-EPR Spectroscopy in Conjunction with *de novo* Protein Folding

Axel W. Fischer[1,2], David M. Anderson[4], Maxx H. Tessmer[5], Dara W. Frank[5], Jimmy B. Feix[3,*], Jens Meiler[1,2,*]

1. Department of Chemistry, Vanderbilt University, Nashville, TN 37232, USA
2. Center for Structural Biology, Vanderbilt University, Nashville, TN 37232, USA
3. Department of Biophysics, Medical College of Wisconsin, Milwaukee, WI 53226, USA
4. Department of Pathology, Microbiology and Immunology, Vanderbilt University School of Medicine, Nashville, TN 37232, USA
5. Department of Microbiology and Immunology, Medical College of Wisconsin, Milwaukee, WI 53226, USA

Supplementary data

Restraint	D_{SL} [Å]	D_{BB} [Å]	Score
629-645	19.4	23.3	-0.67
636-592	26.4	26.4	-0.99
636-645	20.6	16.1	-0.97
636-649	19.8	11.7	-0.83
636-657	20.0	14.2	-0.94
636-672	28.2	-	-
649-672	28.3	-	-

Table S1: ***SDSL-EPR data available for the C-terminal domain of ExoU.*** *Seven SDSL-EPR distance measurements were available. Shown are the observed mean spin-spin distance (D_{SL}), the C_β-C_β distance of the spin-labeling sites in the X-ray-derived model (D_{BB}, Protein Data Bank entry 3TU3), and the agreement score of the X-ray-derived model with the SDSL-EPR data according to the CONE model-based scoring function. Restraints without D_{BB} and score referred to spin-labeling sites not resolved in the X-ray-derived model.*

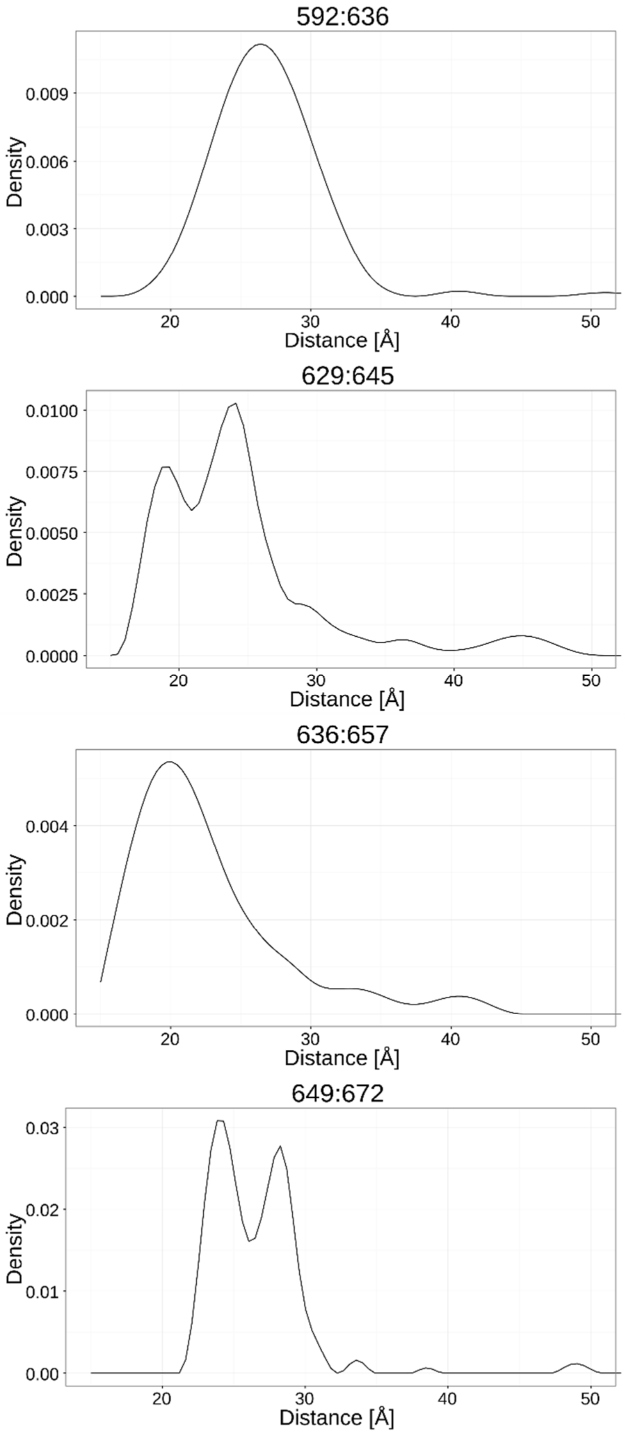

Figure S1: ***Residue-residue distance distributions derived from the DEER-experiment for the C-terminal domain of ExoU.***

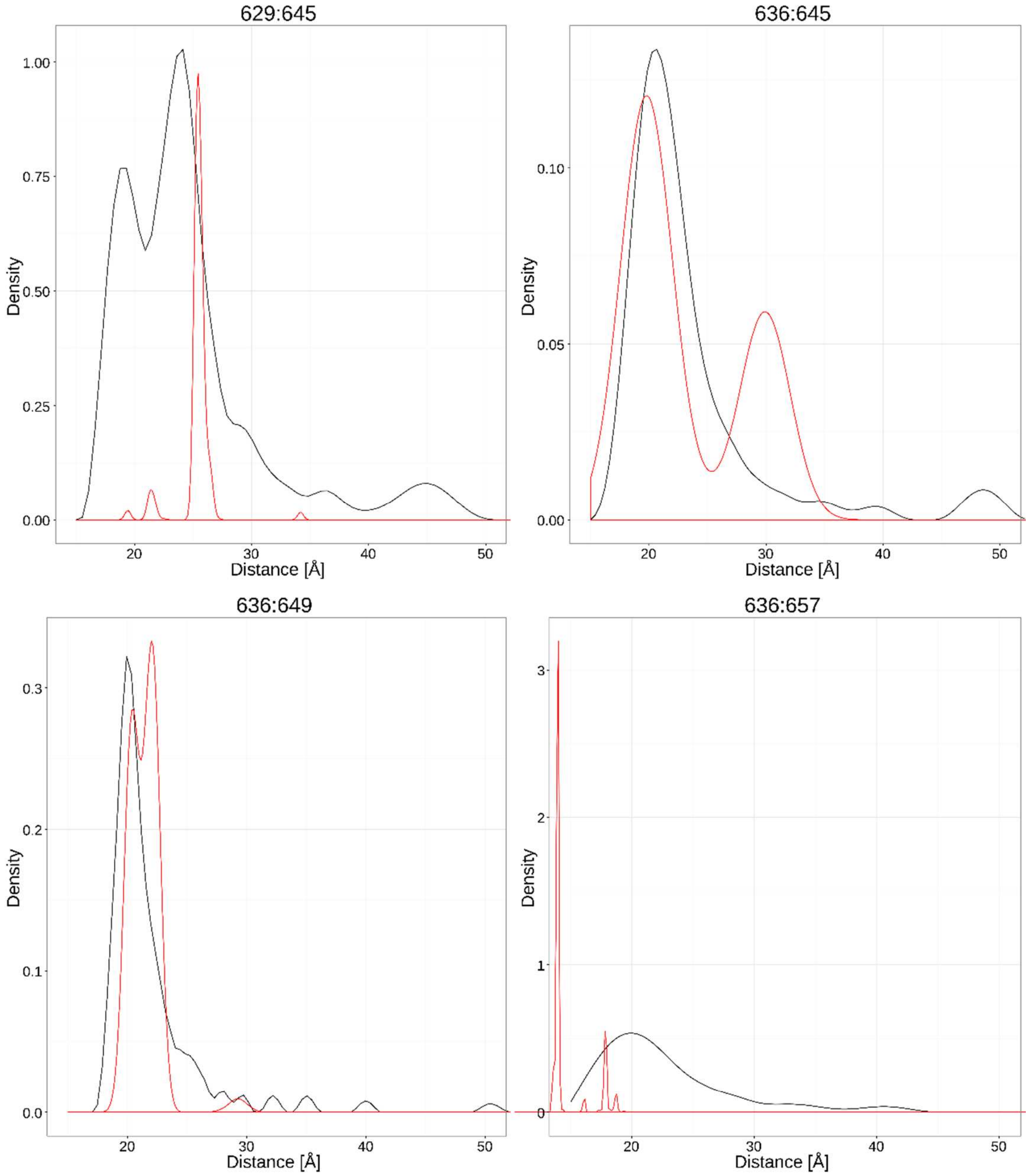

Figure S2: ***Explicit simulation of the spin-labeling pairs used on ExoU.*** *The distance distributions arising from four spin-labeling pairs were simulated explicitly (red) and compared to the experimentally determined distance distribution (black).*

Protocol for the EPR-guided prediction of ExoU

The *de novo* protein structure prediction protocol for ExoU consisted of four modules: secondary structure prediction, topology sampling, clustering, and loop construction in conjunction with high-resolution refinement. The following sections provide a detailed protocol capture to reproduce the data reported in this manuscript.

Procedure S1: prediction of the secondary structure

The secondary structure of ExoU was predicted using PsiPred and Jufo9D. Both approaches are accessible through webservers at http://bioinf.cs.ucl.ac.uk/psipred and http://www.meilerlab.org/index.php/servers/show?s_id=5. The webservers require the primary structure of the protein in question and the predictions' output files are required for topology search module.

From the output files, an SSE pool can be created using the BCL with the following command line:

```
bcl CreateSSEPool -ssmethods JUFO9D PSIPRED -pool_min_sse_lengths 5 3
-sse_threshold 0.5 0.5 0.5 -prefix <prefix> -join_separate -factory
SSPredThreshold
```

Procedure S2: topology sampling

The EPR-derived distance restraints have to be formalized in a BCL-readable format like the following example, which defines two EPR-derived distance restraints:

```
Atom Distance Assigned

A    20      CB  A 40 CB  26.4  100 1

A    56      CB  B 61 CB  20.6  100 1
```

This module is based on the BCL::Fold protocol for soluble proteins and requires the previously created SSE pool and the restraint files. The topology of the protein can then be sampled using the following command line:

```
bcl  -stages_read stages.txt -restraint_types DistanceEPR -
restraint_prefix <cst_prefix> -protein_storage <output_folder> -prefix
<output_prefix> -sequence_data <input_prefix> 3tu3 -sspred PSIPRED -
opencl Disable -nmodels <num_models> -start_model <start_model>
```

The flag for providing a start model was only set for the second iteration of the topology sampling.

Procedure S3: clustering of the sampled models

Clustering was performed using a k-means implementation in R in conjunction with using the RMSD as dissimilarity metric. Once a matrix containing the pairwise dissimilarities between models has been obtained, the clustering can be performed in R using the following sequence of commands:

```
// load libraries
library(cluster)

// load the dissimilarity matrix created with the BCL
data_mat = as.matrix(read.table("distance_matrix.tbl", header = T))

// create a full matrix
data_mat = data_mat + t(data_mat)

// convert into a dissimilarity matrix
data_mat = as.dist(data_mat)
```

```
// cluster for k cluster centers
clusters = pam(data_mat, k)

// this will give you some information regarding the clusters
clusters$clusinfo

// this will give you the medoids of the clusters
clusters$medoids
```

The silhouette score can be directly computed from the clusters object using the command silhouette(clusters).

Procedure S4: loop construction and high-resolution refinement

The high-resolution refinement was performed using the Rosetta software suite and can be performed using the following command line:

```
loopmodel -loops:frag_sizes 9 3 1 -loops:frag_files <fragments_9>
<fragments_3> none -loops:remodel quick_ccd -loops:refine refine_ccd -
loops:extended -loops:relax fastrelax -ex1 -ex2 -database <database> -
nstruct <num_models> -in:file:s <start_model> -loops:loop_file
<loops_file> -out:prefix <output_prefix> -constraints:cst_file
<restraint_file> -constraints:epr_distance -score:weights
<score_weights> -overwrite
```