PROCEEDINGS OF SCIENCE

Accurate Fitting SAXS Curves with NMR Structure Ensembles [*]

Aleš Křenek
Masaryk University
E-mail: ljocha@ics.muni.cz

Karel Kubíček
Masaryk University
E-mail: karel.kubicek@ceitec.muni.cz

Richard Štefl
Masaryk University
E-mail: richard.stefl@ceitec.muni.cz

Jiří Filipovič
Masaryk University
E-mail: fila@ics.muni.cz

Typical NMR analyses of a biomolecule yields a set of up to few dozens candidate 3D structures of the analyzed molecule without any clues to discriminate among them further. A parallel SAXS experiment on the same sample can be used for this purpose.

Previous implementations of "ensemble fit" (search for a mix of molecular conformations which matches the SAXS curve) were designed to choose from a huge ensemble generated by molecular dynamics. Therefore the methods must trade off accuracy for manageable speed, and they end up in mixing curves computed with rather different values of parameters which have physical meaning, which should be avoided.

On the contrary, with a relatively small input set of candidate NMR structures we take a more accurate approach. Both the model parameters, considered globally now, and weights of individual candidate structures (reflecting their presence in the solution) become independent variables of a multidimensional global optimization problem; the optimized value is the accuracy of the fit to the experimental data. The optimization must escape from traps of many local minima therefore we use Monte Carlo with stochastic tunnelling. The method also offers opportunities for parallelization.

The final issue is user friendliness of the entire workflow, which is quite complex, involving several programs to be run, handling different file formats, and setting multiple parameters, ending up with visualization of results. We outline design of a web portal hiding these complexities to the end user.

The International Symposium on Grids and Clouds (ISGC) 2015
15 – 20 March 2015
Academia Sinica, Taipei, Taiwan

PoS(ISGC2015)025

[*] This work was done with the support of Masaryk University, grant no. MUNI/M/1038/2013. Computational resources were provided by the MetaCentrum under the program LM2010005 and the CERIT-SC under the program Centre CERIT Scientific Cloud, part of the Operational Program Research and Development for Innovations, Reg. no. CZ.1.05/3.2.00/08.0144.

1. Fitting SAXS Curves on NMR Structures

A typical output of the analysis of NMR restraints (e.g. run of Cyana program [2]) is a set of at most few dozens candidate 3D structures of the analyzed molecule (protein). However, the analysis does not give information which of those candidates are more favoured, and in what ratio they occur in the solution.

If the SAXS scattering curve is measured on the same sample, this information can be used for the purpose. A particular 3D structure is used to compute a theoretical SAXS curve according to some model. There are several approaches (Sect. 2), however, they are typically based on the Debye formula (e.g. [6]):

$$I(q) = \sum_{i=1}^{N} \sum_{j=1}^{N} f_i(q) f_j(q) \frac{\sin q d_{ij}}{q d_{ij}} \tag{1.1}$$

where N is number of atoms, d_{ij} is their Euclidean distance, and q is transfer momentum (see Sect. 2). The *form factors* f_i, besides characterising the atoms, are expressed in terms of further constants c_1, c_2 which reflect behaviour of the solvent and the solvent accessible area of the macromolecule.

Given this model, fitting the experimental curve is done by minimizing

$$\chi^2 = \frac{1}{M} \sum_{i=1}^{M} \left(\frac{I_{exp}(q_i) - cI(q_i)}{\sigma_i} \right)^2 \tag{1.2}$$

with the least-squares method (M is number of the measured points of the momentum q_i). Further, c_1 and c_2 are swept over appropriate intervals with 20–100 discrete steps. Figure 3 shows a typical shape of the fitted χ wrt. c_1 and c_2.

The fitting is done for each of the NMR candidate structures, and those with the best fit (the lowest χ) are chosen. However, because of treating the conformations independently, such procedure does not give neither any information on their ratio in the solution, nor guarantee they are the prevailing ones (i.e., the "best" fit still can be rather poor).

Due to linearity of Eq. 1.1, a SAXS curve of a mixed solution of several conformations is a linear combination of the curves of individual conformations according to their ratio. Previous implementations of the *ensemble fit* (search for a true mix of conformations which matches the SAXS curve, see Sect. 2 for details) were designed to choose from a huge ensemble generated by molecular dynamics. Therefore the methods must trade off accuracy for manageable speed. Typically, a genetic algorithm is used to generate a candidate subset of conformations. For such a selection a combined SAXS curve is computed, using precomputed model curves for the ensemble members, and it is fitted to the experimental data.

The principal drawback of this approach is mutually independent computation of the model curves. However, it can be seen (Fig. 4) that the χ-minimizing values of c_1 and c_2 differ significantly for different conformations. While the constants have physical meaning unrelated to a particular conformation, and we are trying to describe behaviour of a mixed sample, consistent values of c_1 and c_2 must be used instead.

We checked the FoXS-based ensemble fit (Sect. 2) with our reference protein (161 residues, 2549 atoms) and several other datasets. It appears to yield ensembles of conformations with quite different c_1 and c_2 values (with our protein -0.20 vs. 0.81 in c_2 for two top-scoring conformations).

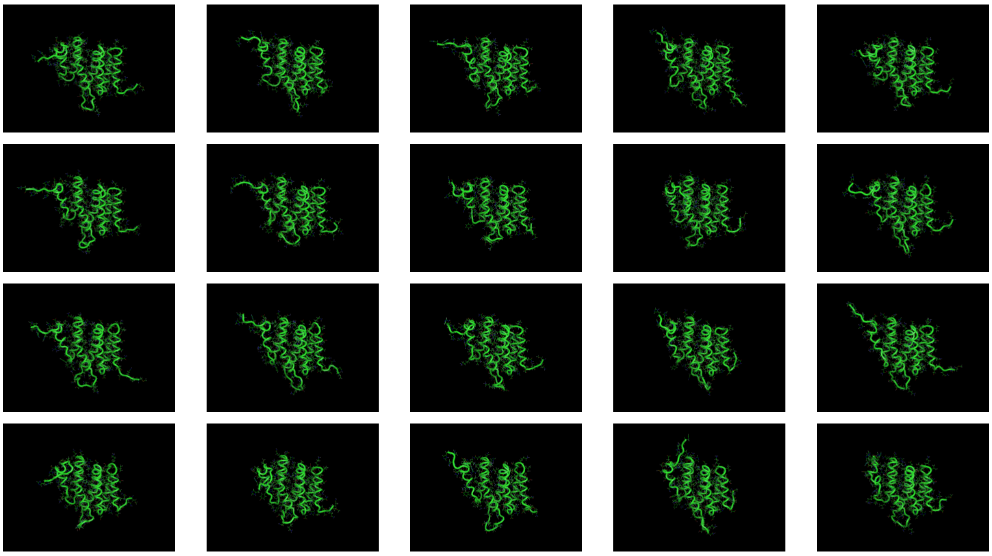

Figure 1: NMR candidate shapes of our reference protein

2. Related Work

NMR[1] is a well known experimental method in structural biology. Processing of the experimental data, which is quite complex task, leads to a set of probable shapes of the molecule finally. In our work we mostly use the CYANA [2] software, Fig. 1 shows a typical example.

SAXS is another experimental method which exposes a sample to X-ray and it measures the ray scattering. The experiment yields *scattering curve*, a function of the ray intensity depending on the scattering angle θ, usually expressed as the momentum transfer $q = 4\pi \sin\theta/\lambda$, which makes the curve independent on the actual ray wavelength λ.

On the other hand, given a fixed molecular structure, an expected scattering curve can be computed and compared with the experiment. A good overview of the combined methods is given in [5]. In particular, we are interested in the curve computation and fitting. The original approach was introduced in [7] with the CRYSOL software, and later refined several times, e.g. [4]. Sligtly different approach, FoXS software using another model of interaction of the protein with solvent, was proposed in [6]. A typical output is shown in Fig. 2.

The problem of fitting ensembles of conformations to a SAXS curve is addressed in [3], however, the work aims in different direction—choosing from huge ensemble (generated by MD), the global optimization uses genetic algorithm, and the protein-solvent interaction parameters are computed with independent runs of FoXS (see Sect. 3). Similar approach (based on CRYSOL) was presented in [8, 1]. Both the systems are provided to the community as web portals[2].

3. The Method

In order to overcome the principal drawback of previous methods described above, we define

[1] E.g. `http://en.wikipedia.org/wiki/Nuclear_magnetic_resonance`

[2] `http://modbase.compbio.ucsf.edu/foxs/`, `http://www.embl-hamburg.de/biosaxs/online.html`

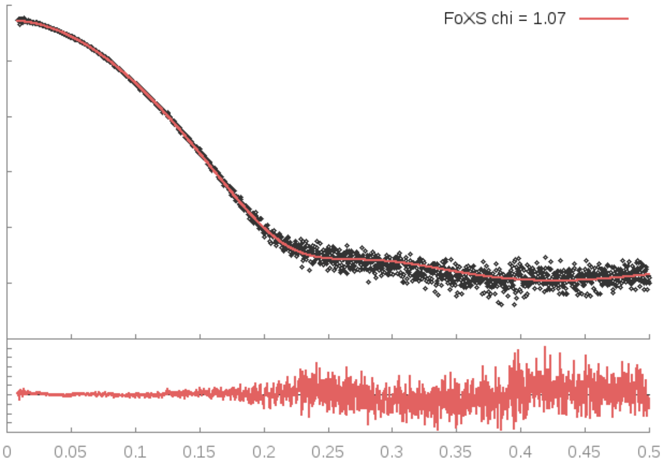

Figure 2: Fitting SAXS data to one of the protein shapes above. x axis spans q up to 0.5 A^{-1}, y axis is $\log I$ (absolute scale is irrelevant)

the mixed expected curve of S different conformations as

$$I(q) = \sum_{i=1}^{S} w_i I_i(q, c_1, c_2) \qquad \text{with} \quad \sum_{i=1}^{S} w_i = 1 \tag{3.1}$$

where c_1, c_2 are common solvent parameter values, and w_i are relative weights of the conformations—their ratio in the solution. This $I(q)$ is used in Eq. 1.2 for the least-square fit to the experimental data.

With typical NMR data we have 10–50 different conformations (S). Finding the best ensemble fit becomes a global optimization problem in $S+2$ variables (w_i, c_1, c_2) under the constraint $\sum_{i=1}^{S} w_i = 1$.

Varying location of partial minima wrt. c_1, c_2 (Fig. 4) for individual conformation yields the function to be minimized to have very high number of shallow local minima, making the problem to be true global minimization with no a priori clues where to search for the minimum.

Therefore we choose a Monte Carlo method. In each step a random change vector in w_i, c_1, c_2 of length up to 10^{-2} (found experimentally) is generated, it is constrained to keep the condition $\sum w_i = 1$, and it is added to the current w_i, c_1, c_2. The result is accepted according to Metropolis criterion tuned to accept increase in χ^2 by 10^{-3} with the probability of 10 %.

Further we improve convergence of the Monte-Carlo search with stochastic tunnelling[3]—instead of χ^2 the Metropolis criterion is applied to

$$1 - e^{-\gamma(\chi^2 - \chi^2_{\text{min}})} \tag{3.2}$$

where χ^2_{min} is the best value found so far. (The method fills in local minima higher than χ^2_{min} while it deepens lower ones.)

The approach might lead to overfitting—the number of non-zero w_i's is not restricted, therefore arbitrary number of could contributed to the combined curve, yielding an unrealistic “best” fit.

[3] E.g. `http://en.wikipedia.org/wiki/Stochastic_tunneling`

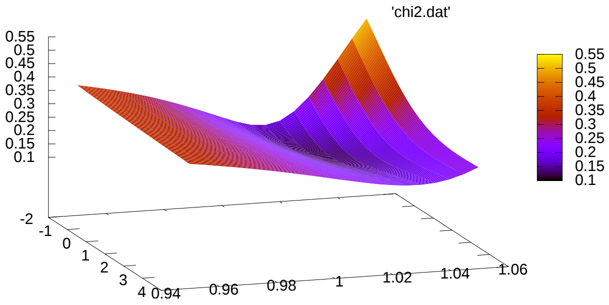

Figure 3: Typical dependence of the fitting score (χ) on solvent parameters c_1, c_2

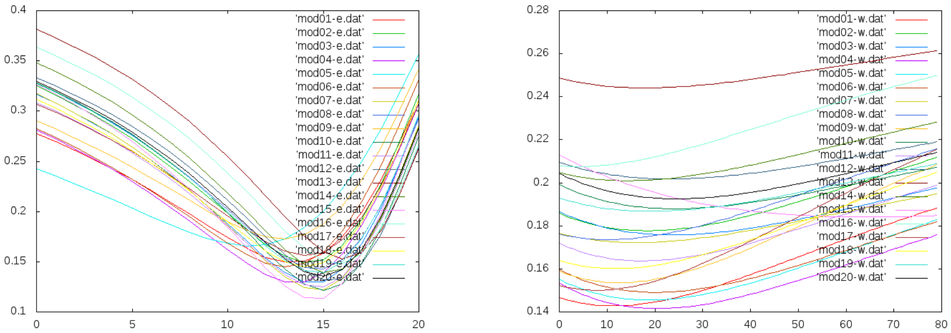

Figure 4: Dependence of χ on c_1 and c_2 solvent parameters (the other one is fixed) for 20 different conformations of our reference protein. x axes span over feasible ranges of the parameters ($c_1 \in [0.95, 1.05], c_2 \in [-2, 4]$)

However, experience shows that it is not the case with real NMR structures. The minimization converges to a set of only few non-negligible weights, eliminating the others, regardless on its starting point.

4. Implementation

A straightforward implementation of the method described above would require computation of the curve fitting, for fixed c_1, c_2 to be done S (number of structures) times in each optimization step. Single such computation takes approx. 200 ms. According to our experience the optimization must run 10^7 steps in order to be sure the rich surface was exhausted and the global minimum found. Therefore for e.g. $S = 20$ we get $10^7 \times 20 \times 200$ ms, i.e. 1.26 year running time, which is definitely not acceptable.

The problem can be overcome by independent precomputation of the curves for each conformation and sufficiently fine sampling of c_1, c_2, and interpolating them linearly in the optimization. Because for a single conformation the χ function is not too rich in extrema (Fig. 3).

4.1 Components

The whole computation is a sequence of three essential steps:

1. Precompute theoretical SAXS curves for running values of c_1, c_2.

   For each of the candidate structures (PDB file), and for the parameters running over a reasonable range (we use $c_1 \in [0.95, 1.05], c_2 \in [-2, 4]$) in sufficiently fine steps (0.005 for c_1 and 0.05 for c_2) the curve is computed using the publicly available FOXS program [6]. The experimental profile is used to fit the theoretical curve to a globally consistent scale.

2. Compact results of the previous step.

   Given the parameter range, the above computations produce 2541 text files of approx. 80 kB each for each of the candidate structures (approx. 200 MB of data). Reading and parsing those files become tedious, especially when the next step is run repeatedly. Therefore we parse them and store in a single-purpose binary format, only one file (approx. 35 MB) per each structure.

3. Global optimization of the curve fitting.

   This steps implements the method described in Sect. 3 with a C++ program. The code is fairly simple (approx. 1,500 lines only). The stochastic optimization itself is implemented in a generic class, which provides methods of generating random steps and bookkeeping the system state. The class is extended further with specific implementations of step-acceptance strategy (besides the stochastic tunnelling we provide pure Monte Carlo).

   Another class is responsible for managing the input data (multiple theoretical curves read from the binary files, see above). This class also implements the curve interpolation – given arbitrary c_1, c_2 values (as required by the optimization step), four closest curves computed in the first step are found, and they are linearly interpolated to give an approximation for the specific parameter values. Because it turns out that for a fixed structure χ as a function of c_1, c_2 resembles a paraboloid (Fig. 3), the approximation is acceptable.

4.2 Sequential performance and parallel design considerations

Our overall goal is achieving as much (semi-)interactivity of the cloud deployment of the computation. Besides optimizing individual pieces of code, the principal instrument is parallelization. In this section we discuss the opportunities. Because of the Amdahl's law all the three steps must be evaluated.

The measuremets we refer to bellow were done on the *Zapat* cluster of CERIT-SC[4], having E5-2670 2.60GHz CPUs and large enough memory to mask I/O overheads effectively. The testing data were the results of Cyana run on the CID protein (161 residues, 2549 atoms) and SAXS data having 1794 measurements.

All the measurements were repeated 5 times, and minimal times are reported to give the lower bound while avoiding interference effects of the shared environment.

[4] `http://www.cerit-sc.cz/en/Hardware/`

Curve precomputing	2541 invocations of FOXS	8’03”
Compacting results	processing 2541 curve files	0’20”
Optimization	10^7 steps of stochastic tunnelling	32’12”

Table 1: Times of sequential steps

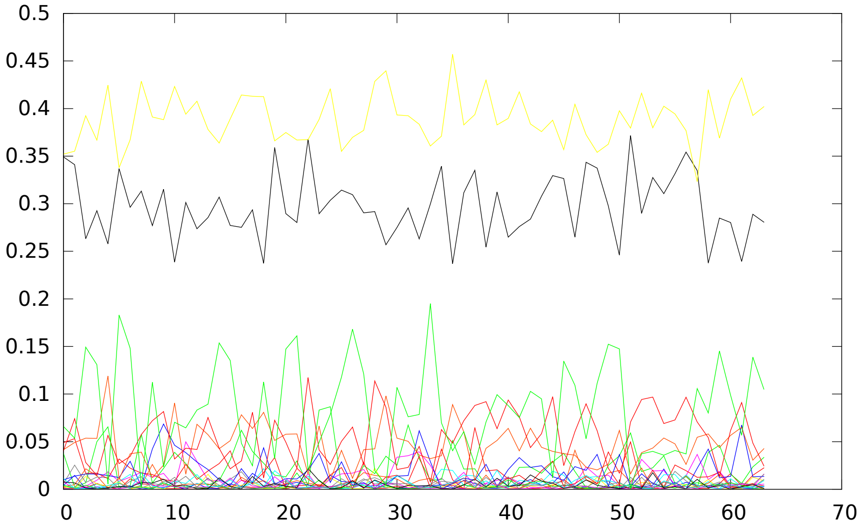

Figure 5: Raw results of the stochastic tunneling run from 64 different starting points (horizontal axes). The lines represent resulting weights of the 20 structures as they differ in each result. They are two dominant structures (approx. 0.30 and 0.39), three other contributing, and the rest is negligible.

The measurement results are shown in Tab. 1. Depending on the starting point, we observed the optimization to keep improving the minimum for very long time, approx. 1.9–9.2 × 10^7 steps. However, the decrease tends to be exponential, with the most of the improvement achieved in the first 10 % of the computation. Therefore we consider 10^7 to be the safe limit for reaching an acceptable approximation of the global minimum.

Moreover, we ran the optimization from multiple different starting points anyway (see bellow), and all the achieved minima were quite close to each other. This indicates a shallow basin on the function hypersurface was reached; running the optimization further would just oscillate in this basin, not bringing additional benefits. (Fig. 5 shows how the results among 64 independent runs differ.

We consider 10^7 steps to be a reasonably safe upper bound to reach an acceptable minimum. Despite there is a room for improvement in the curve precomputing, the optimization is still dominating, therefore we can focus on it for the time being.

4.3 Parallel stochastic tunnelling

The optimization algorithm is stochastic, therefore we expect its rate of convergence to depend strongly on the starting point—as we have no clues, it is the matter of good luck how fast is the global minimum found.

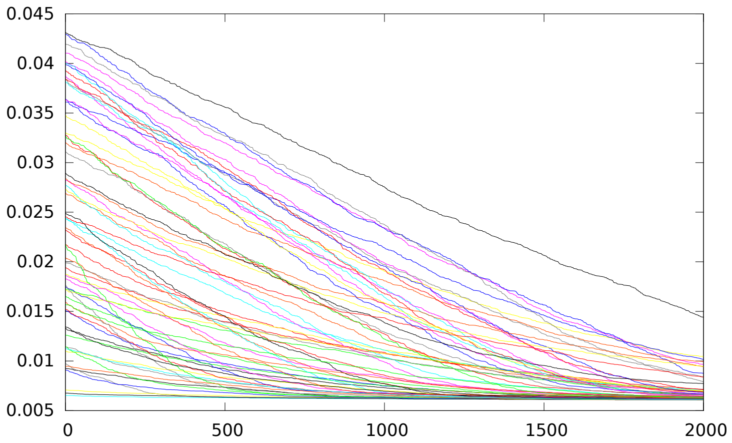

Figure 6: Progress of optimization with 64 different starting points. Horizontal axis is numbered in thousands of steps.

To test this hypothesis, we ran the computation with 64 different randomly generated starting points. Fig. 6 shows the progress of those computations. Apparently, even the starting points differ significantly, the optimization converges in all cases but its progress is more or less random. In particular, a better initial guess does not imply faster convergence in general.

Therefore principal gain of parallelization is stochastic again—while keeping the number of steps (hence running time) constant and reasonable, the more instances we run, the more likely we find the global minimum.

A possible improvement is a regular exchange of the “best so far” minimum among the processes—the intermediate results of the other process affect acceptance of the Monte-Carlo steps according to the modified criterion (Eq. 3.2). However, the practical gain of this extension is questionable. In all our experiments, the winning process was completely determined by the initial choice of starting point, and even revealing the intermediate results didn’t help the others to find a better solution.

5. Cloud Deployment

We ended up with the implementation consisting of a set of bash scripts which can be run to compute the scattering curves of all the conformations and c_1, c_2 samples in parallel in a few minutes, and a C++ MPI program to do the global optimization in less than half an hour. Both can be easily submitted to a traditional batch system of a computing centre.

However, this is still not the interface the users expect nowadays. Their understanding of “the Cloud” is Software-as-a-Service in this case. They expect a simple web portal to upload a PDB file with the structures and the experimental data, to choose from a few options eventually, to spawn the computation and to collect the results, preferably presented in a graphical form whenever appropriate.

Currently we are developing a web portal along these lines. As the principal results of the computation is the vector of weights of the conformations, the interface allows the user to set a threshold value—lower weights are considered to be negligible, and only conformations above the threshold are displayed as 3D model as well as included in the displayed curve fitting.

The portal keeps a dynamic pool of CPU cores available for the computation, depending on the actual number of user requests submitted and the load of the infrastructure behind. In this way a semi-interactive response is achieved, even for rather heavyweight computations.

6. Conclusions

Combined analysis of NMR and SAXS experimental results on the same sample turns to be beneficial—fitting the SAXS curve with an ensemble of conformations allows refinement of the NMR results which don't give hints on the actual presence of candidate conformations otherwise.

We propose a computational method which improves the previous implementations of ensemble fit by unconstrained global optimization of the fit, allowing any mixture of the candidate conformations. Our experiments confirm that the method does not suffer from overfitting; on the contrary it discriminates among the candidates quite well.

With a parallel implementation, the method yields trustworthy results in a reasonable fixed computing time. Therefore it is suitable to be presented to the user as a SaaS cloud implementation, with easy-to-use web interface, giving semi-interactive response, and managing the computing resource allocation transparently.

References

[1] P. Bernado, E. Mylonas, M.V. Petoukhov, M. Blackledge, and D.I. Svergun. Structural characterization of flexible proteins using small-angle X-ray scattering. *J. Am. Chem. Soc.*, 129(17):5656–5664, 2007.

[2] P. Güntert, C. Mumenthaler, and K. Wüthrich. Torsion angle dynamics for NMR structure calculation with the new program DYANA. *Journal of Molecular Biology*, 273(1):283–298, 1997.

[3] M. Pelikan, G.L. Hura, and M. Hammel. Structure and flexibility within proteins as identified through small angle X-ray scattering. *Gen Physiol Biophys.*, 28(2):174–89, 2009.

[4] Maxim V. Petoukhov and Dmitri I. Svergun. Global rigid body modeling of macromolecular complexes against small-angle scattering data. *Biophysical Journal*, 89(2):1237–1250, 2005.

[5] Christopher D. Putnam, Michal Hammel, Greg L. Hura, and John A. Tainer. X-ray solution scattering (SAXS) combined with crystallography and computation: defining accurate macromolecular structures, conformations and assemblies in solution. *Quarterly Reviews of Biophysics*, 40:191–285, 8 2007.

[6] Dina Schneidman-Duhovny, Michal Hammel, John A. Tainer, and Andrej Sali. Accurate SAXS profile computation and its assessment by contrast variation experiments. *Biophysical Journal*, 105:962–974.

[7] D. Svergun, C. Barberati, and M.H.J. Koch. CRYSOL – a program to evaluate X-ray solution scattering of bilogical macromolecules from atomic coordinates. *J. Appl. Cryst.*, 28:768–773, 1995.

[8] G. Tria, H.D.T. Mertens, M. Kachala, and D.I. Svergun. Advanced ensemble modelling of flexible macromolecules using X-ray solution scattering. *IUCrJ*, 2:207–217, 2015.