Computers & Geosciences 93 (2016) 63–69

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Case study

Rock.XML – Towards a library of rock physics models

Erling Hugo Jensen [a,b,*], Ragnar Hauge [c], Marit Ulvmoen [c], Tor Arne Johansen [a,d], Åsmund Drottning [e]

[a] Department of Earth Science, University of Bergen, Bergen, Norway
[b] Rock Physics Technology, Bergen, Norway
[c] Norwegian Computing Center, Oslo, Norway
[d] The University Centre in Svalbard, Norway
[e] NORSAR, Kjeller, Norway

ARTICLE INFO

Article history:
Received 22 September 2015
Received in revised form
21 April 2016
Accepted 22 April 2016
Available online 23 April 2016

Keywords:
Rock physics models
Reservoir geophysics
XML

ABSTRACT

Rock physics modelling provides tools for correlating physical properties of rocks and their constituents to the geophysical observations we measure on a larger scale. Many different theoretical and empirical models exist, to cover the range of different types of rocks. However, upon reviewing these, we see that they are all built around a few main concepts. Based on this observation, we propose a format for digitally storing the specifications for rock physics models which we have named Rock.XML. It does not only contain data about the various constituents, but also the theories and how they are used to combine these building blocks to make a representative model for a particular rock. The format is based on the Extensible Markup Language XML, making it flexible enough to handle complex models as well as scalable towards extending it with new theories and models. This technology has great advantages as far as documenting and exchanging models in an unambiguous way between people and between software. Rock.XML can become a platform for creating a library of rock physics models; making them more accessible to everyone.

1. Introduction

Rocks are solid aggregates of minerals, with possible traces of organic material and various porosities. This void space is filled with fluid or mixtures of different fluids; such as brine, gas and oil. Rock properties vary greatly due to the large number of possible constituents and compositions. For example, some rocks are unconsolidated and very loose, while others are more dense, consolidated and cemented. Even details such as if the cement is coating the grains or located at the contact points between the grains, will affect the stiffness of a rock (Dvorkin et al., 1994).

There is a wide range of theories for modelling all these different types of rocks and a review of some of them is given by Avseth et al. (2010). However, new ones and new ways of combining old theories are continuously emerging. But all modelling typically share a common workflow:

1. Identify representative solid and fluid constituents.
2. Specify constituent properties, either
   a. from tabulated values found in the literature, or
   b. calculate them
3. Calculate the effective mineral properties by combining the solid constituents.
4. Calculate the effective fluid properties by combining the fluid constituents.
5. Calculate the porous dry rock properties for a given porosity.
6. Calculate the effective fluid saturated rock properties for that porosity.

Some variations can of course occur, e.g. steps 5 and 6 are sometimes done simultaneously. Rock physics theories are applied for the various calculation steps, and the choice of theories is closely linked to the type of rock we have. Hence, interpretation in reservoir geophysics starts when choosing which rock physics model and theories to use.

We propose a format for uniquely specifying rock physics models. Hence, we can digitally store data about the various rock constituents and other relevant parameters. In addition, we store details on how the rock is composed, i.e. how to do the actual modelling by specifying which theories to use and how they should be combined. We have named this format Rock.XML and it is based on the extensible markup language XML. We have chosen

* Correspondence to: Løvåsbakken 21, H0103, 5145 Fyllingsdalen, Norway.
E-mail address: sschwss@hotmail.com (E.H. Jensen).

http://dx.doi.org/10.1016/j.cageo.2016.04.011
0098-3004/

the XML platform because it has a simple set of basic rules, yet a high degree of flexibility which makes it very easy to define custom markup languages for storing data. Furthermore, it supports a hierarchical structure which we actively use in the specification of the rock composition. (See Learning XML by Ray (2003) or XML in a Nutshell by Harold and Means (2004) for more details about XML).

XML was created as a counterpart to the Hypertext markup language (HTML). While HTML focuses on how to present data, XML focuses on what the data is (Ray, 2003). The use of XML within geoscience is not new. For example, Houlding (2001) discussed the potential for using it to store data, and Mello and Xu (2006) explain how XML can improve the efficiency of application development in geosciences. It is also possible to use XML to automate the transformation of data between different geoscience data standards (Nance and Hay, 2005). A review of the application of the geography markup language (GML) within the field of geology was given by Lake (2005), and Sen and Duffy (2005) propose another markup language which extends GML to have a wider geoscience scope. Another extension to GML was designed by Nativi et al. (2005) to integrate the use of the Network Common Data Form (netCDF). Babaie and Babaei (2005) discuss the use of XML Schema and namespaces in modelling geological objects. An online reservoir modelling software, which makes use of XML-based data handling procedures, has been created by Victorine et al. (2005). XML has also been used to describe details about rock mechanics tests such as uniaxial compression experiments (Li et al., 2012), i.e. details about the rock materials, test preparations and procedures. But we believe this is the first time it is used for storing details about rock physics models.

There are several applications for Rock.XML. It can be used in documenting applied models and specifications for a particular modelling process. This is useful not only to the individual researcher, but also for other researchers. It simplifies communication and assures collaborating colleagues are working on the same models throughout a project. The same is the case when publishing results in a journal; the relevant models can be available in Rock.XML file format as downloadable extra material – just as we are doing for this paper. Furthermore, it is possible to create software algorithms which read the files, parses the content and performs the specified modelling directly. This opens up several possibilities for modelling software such as loading and storing models to files, importing/exporting models between other software and initiating different types of batch jobs. It can be used in forward and inverse rock physics modelling, e.g. for generating constraint cubes (Johansen et al., 2013) or for generating rock physics models for use in seismic inversion (Kolbjørnsen et al., 2013) The close connection between XML and the World Wide Web invites the use of these files in various ways through browser based user interfaces or even so-called smart phone apps. It would be possible to create a library of rock physics models which are accessible online and updated with new ones as soon as they are proposed.

In this paper, we focus on the philosophy and general idea behind Rock.XML. Therefore, we show only a few examples with XML code to illustrate how it can be implemented, and instead use flowcharts to communicate our philosophy. We have included three modelling examples; for an unconsolidated friable-sand, a cemented sandstone and finally an example for a consolidated sandstone using an inclusion based model. Rock.XML files for the three rock physics models, together with a manual for the Rock.XML markup language can be downloaded as extra materials.

2. Basics of the Rock.XML format

The XML format has a set of simple rules built around the concept of defining elements. An element, identified by a tag name and with possible associated attributes, can contain data and/or other sub elements; i.e. the possibility of creating nested structures. This nesting is quite useful in rock physics modelling, where a saturated rock is often built from fluids and solids, which again may be created from mixing more fundamental constituents. The XML format is well suited for storing such recursive patterns.

The fundamental building block in an XML file is the tag, with its corresponding closing tag. Between these, there may be other tags, or element values. Consider the following example:

```
<variable>
  <label>pore pressure</label>
  <value>10</value>
</variable>
```

Here we have the two tags “label” and “value”, which both have immediate element values. They are grouped together under a “variable” tag, to mark that they belong together. This simple, yet very flexible structure is an ideal platform to define various markup languages tailored to specific tasks or problems. We refer to our markup language as Rock.XML. The language is defined by the tags it accepts, and how these relate hierarchically to each other.

The 6-step modelling workflow provided in the introduction, shows that rock physics modelling is a process of defining building blocks and applying theories to combine them. The result can either be a new building block or a representation of the effective rock. As such, we have defined Rock.XML around the concept of constituents, elements that either may be an end result, or a part of a more complex model. We use the following constituents: *solid*, *fluid*, *dry rock* and *rock*, which becomes our basic tags. In a programming setting, these can be considered as objects with associated attributes.

A block is the set of tags and values between a tag and its closing tag. In each constituent block there is a `<label>` element, giving a unique identification (or instance name to an object) for referring to them in multiple places in the modelling, and help with the understanding of what is being defined. Furthermore, there is also always a theory block, giving a theory that defines the constituent, and its parameters. We will illustrate this by giving simple examples of our constituents.

2.1. The solid constituent

A *solid* has three associated attributes; namely bulk modulus, shear modulus and density. They are the relevant mineral properties for the modelling we want to do. This means that the information given for a solid must tell us how to compute these. Below is an example where we specify the properties for quartz (Mavko et al., 2009).

```
<solid>
  <label> quartz </label>
  <tabulated>
    <bulk-modulus> 37 </bulk-modulus> <!- GPa ->
    <shear-modulus> 44 </shear-modulus> <!- GPa ->
    <density> 2.65 </density> <!- g/cm^3 ->
  </tabulated>
</solid>
```

Here we have used `<tabulated>` as our theory. This is in reality no theory as such, but a method to set each of the attributes individually; either by value as is done here, or reference to a variable which in turn could be a result of some type of modelling. The `<!- text ->` is how comments are written in XML.

2.2. The fluid constituent

A *fluid* is a special case of a *solid*, where the shear modulus is set

to 0 GPa because a fluid has no shear resistance. Hence, a tabulated specification of a *fluid* would look similar to the example above for a *solid*, except the <shear-modulus> element is omitted. Below is another example of how a fluid can be defined. Here, we use the *theory* of Batzle and Wang (1992) to calculate the bulk modulus and density for brine given its pressure, temperature and salinity.

```
<fluid>
  <label> brine </label>
  <batzle-wang-brine>
    <pore-pressure> 10 </pore-pressure> <!- MPa ->
    <temperature> 60 </temperature> <!- °C->
    <salinity> 0.05 </salinity> <!- fraction of 1->
  </batzle-wang-brine>
</fluid>
```

This theory requires the pore pressure, temperature and salinity in order to compute the bulk modulus and density. Hence, these are given as parameters to the theory which here is specified with the <batzle-wang-brine> tag.

2.3. The dry rock building block

A *dry rock* has bulk modulus, shear modulus, density and porosity attributes associated with it. As the name of this building block implies, the pores in a *dry rock* are assumed to be empty, i.e. not containing any fluids. Below is an example where we use the *theory* of Walton (1987) to calculate the effective properties of a porous unconsolidated rock at critical porosity with smooth surface contacts between the grains.

```
<dry-rock>
  <label> porous sandstone </label>
  <walton>
    <solid>
      <use> quartz </use>
    </solid>
    <coord-nr> 7 </coord-nr>
    <pressure> 10 </pressure>
    <friction> 0 </friction>
    <porosity> 0.23 </porosity>
  </walton>
</dry-rock>
```

The Walton theory for modelling dry rock is based on having a solid, specified with the <solid> tag. The solid might be defined directly, or a predefined solid can be used, as we have done in this example. Again, the other relevant parameters are given.

2.4. The rock

A *rock* may be a building block, but is also the final composition of other building blocks; i.e. typically representing a fluid saturated rock. When using the *theory* of Gassmann (1951), *rock* can be defined as follows in Rock.XML

```
<rock>
  <label> effective saturated rock </label>
  <gassmann>
    <dry-rock>
      <use> sandstone </use>
    </dry-rock>
    <fluid>
      <use> brine </use>
    </fluid>
  </gassmann>
</rock>
```

Note that the porosity and mineral properties, which are required to do the Gassmann fluid substitution, are provided by the *dry-rock* building block.

2.5. Specifying values

In the examples so far, we have explicitly given numbers for each numerical parameter. However, there are a few more options. One is that instead of giving an exact value, a distribution may always be given instead. This is a way of incorporating uncertainties into the model parameters. For instance, in the solid above, the bulk modulus was given by

```
<bulk-modulus> 37 </bulk-modulus> <!- GPa ->,
```

but if we instead wanted a stochastic rock physics model, we could give it as

```
<bulk-modulus>
  <gaussian>
    <mean> 37 </mean> <!- GPa ->
    <variance> 22 </variance>
  </gaussian>
</bulk-modulus>
```

which means that the bulk modulus has a Gaussian distribution with mean 37 GPa and variance 22. Note that the parameters in a distribution cannot be distributions.

Furthermore, wherever a number or text is used as input parameter, it may be substituted by a variable. Variables are predefined values, ensuring consistency when used several places in the code and easier to manage if they need to be tweaked or updated. A variable is defined by a <variable> tag, and takes a label and a value. Here is an example where we have defined a pore pressure variable:

```
<variable>
  <label> pore pressure </label>
  <value> 10 </value> <!- MPa ->
</variable>
```

The <label> is the name we use for referring to this *variable*, <value> can be a number or a text depending on the type the corresponding parameter has. A variable used for the pore-pressure in the earlier *fluid* specification, would change

```
<pore-pressure> 10 </pore-pressure> <!- MPa ->
```

to

```
<pore-pressure>
  <reservoir-variable> pore pressure </reservoir-variable>
</pore-pressure>.
```

Instead of giving a value, we give the reservoir-variable tag, and then the name we assigned to that variable. Variables can of course also be stochastic.

We have decided not to implement units and unit conversions in the format so far, so units are only provided as XML comments. This means that the numbers given will be put unscaled into theories. This gives a flexibility in unit choices, as the choice of input units determines the output unit values, but it requires that all input values are given in consistent units.

2.6. The flexibility of theories

We have already seen some examples of *theories* used in Rock.XML; Batzle and Wang (1992), Walton (1987) and Gassmann (1951). The idea is that the theories can be used to compute the moduli and densities of different constituents. It is easy to add new theories to this format, since each theory defines which parameters it requires. Also note that we do not restrict *theories* to be only rock physics models; any block of information that tells us how to find the moduli and density of a constituent is allowed, as shown in the solid case where we used <tabulated>.

In the following, we will not show any more XML code, but instead illustrate the modelling using flowcharts. Fig. 1 shows an example of a flowchart for modelling a patchy saturation fluid mixture of brine and gas approximated using the theory of Voigt

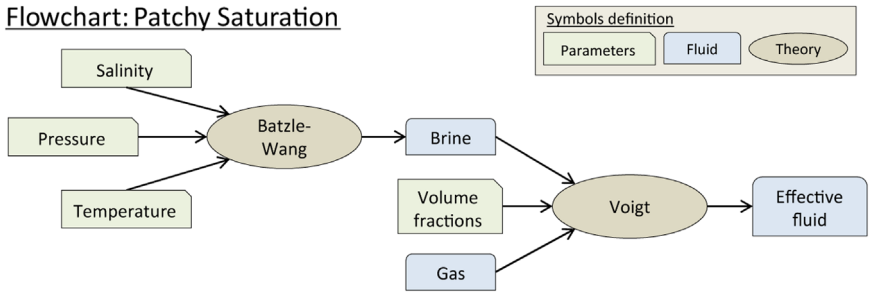

Fig. 1. Flowchart for modelling the bulk modulus and density in a patchy mixture of gas and brine approximated using the theory of Voigt (1928). The brine properties are calculated using the theory of Batzle and Wang (1992), and the gas properties are provided as tabulated values.

(1928). It includes the flowchart for modelling the fluid properties of brine, which we showed the Rock.XML code for in an earlier example. The gas properties are meant to be provided using tabulated values, which we do not explicitly specify in the flowchart.

3. Examples

We have selected three different types of rocks to model;

1) An unconsolidated and non-cemented sandstone.
2) A cemented sandstone.
3) A consolidated sandstone.

We use a friable-sand model (Dvorkin and Nur, 1996) for the unconsolidated sandstone; i.e. a high porosity end-member is modelled using the theory of Walton (1987), then the porosity effects due to sorting is modelled using Hashin-Shtrikman-Walpole lower bounds (Walpole, 1966a, 1966b) and finally the fluid phase is introduced using the theory of Gassmann (1951). For the cemented sandstone we use the constant cement model of Avseth et al. (2000) which is a combination of the contact cement model and the friable-sand model (Dvorkin et al., 1991, 1994; Dvorkin and Nur, 1996). Here we have chosen the effective solid to also be the cementation material, and that the cementation is located at the grain contact points. The consolidated rock is modelled using a differential effective medium theory (Berryman, 1992, 1995) where the host material is the solid matrix and the fluid phase acts as the inclusion material.

The details of the models and parameters in this modelling are given in the Rock.XML files which are downloadable as extra materials. These specifications are illustrated through flowcharts in Figs. 2, 3 and 4. Predicted bulk and shear modulus versus porosity variations for the three models are plotted in Figs. 5 and 6. The density can just as easily be calculated following these specifications, allowing us to proceed with calculating other properties such as acoustic velocities or impedances, Poisson's ratios, etc.

4. Discussion

One goal with Rock.XML is to create a format for documenting rock physics models in an unambiguous way. This is a goal for the XML file format as well (Ray, 2003), making it an ideal platform to build on. An experienced geophysicist might have full control over how the various models are used, but a novice to rock physics

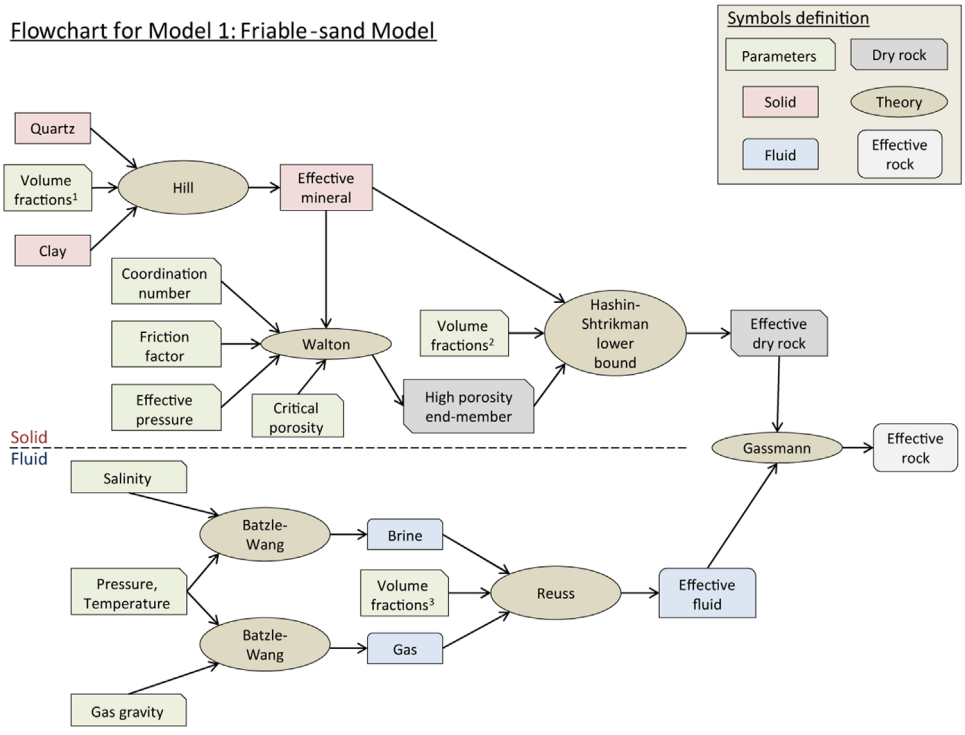

Fig. 2. Flowchart for modelling a rock using a friable-sand model. The volume fraction with indices 1 through 3 can be related to lithology, total porosity and gas saturation, respectively.

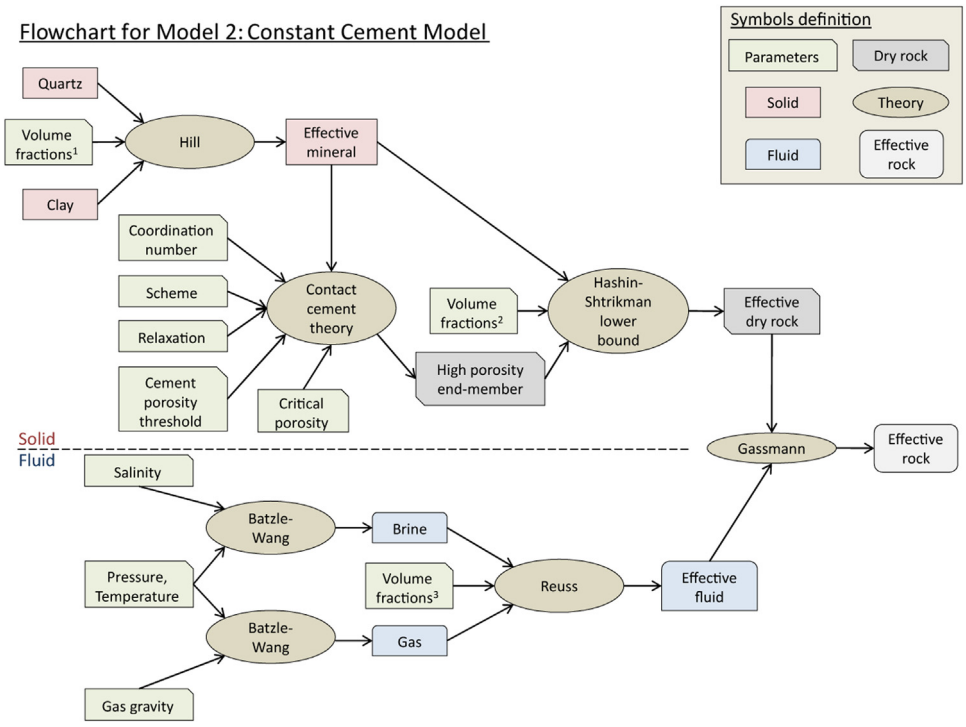

Fig. 3. Flowchart for modelling a rock using the constant cement model. The volume fraction with indices 1 through 3 can be related to lithology, total porosity and gas saturation, respectively.

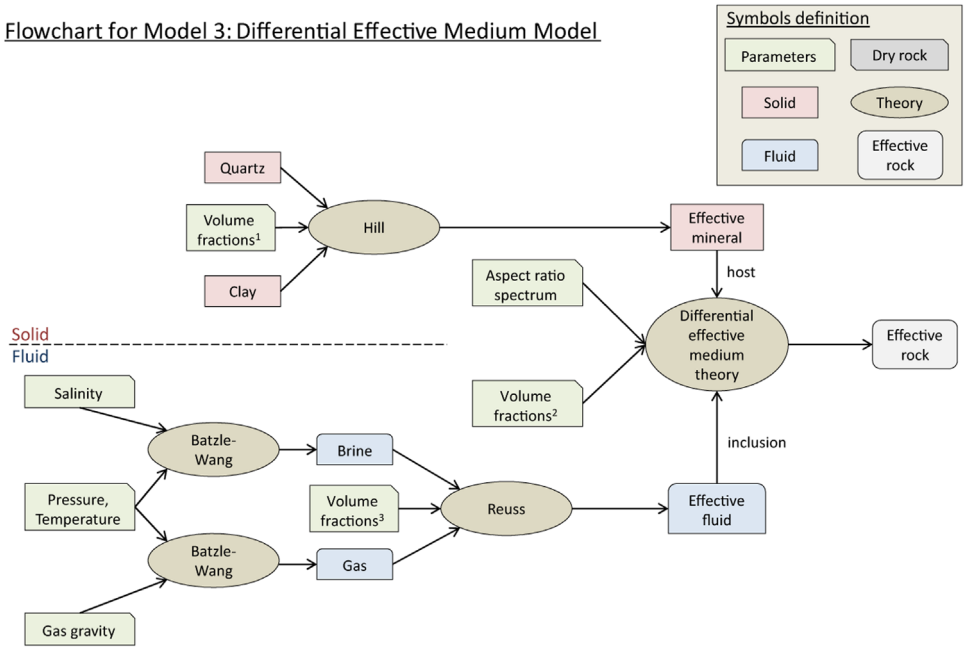

Fig. 4. Flowchart for modelling a rock using the differential effective medium theory. The volume fraction with indices 1 through 3 can be related to lithology, total porosity and gas saturation, respectively.

modelling can have some concerns about the implementations of certain models. Also, some newer models combine various theories such as the patchy cementation model of Avseth et al. (2012) or the "kite shaped model" of Avseth et al. (2014). The finer details of them can take time to decipher for even an experienced geophysicist. Rock.XML will make models more accessible to geoscientists who are not so experienced with the various theories as well as those working with them on a daily basis.

XML is widely supported among various programming languages; making it a lot easier to build so-called parsers for reading and interpreting the data into a software programme. It was specifically designed not only to store data, but also metadata (i.e. data about data). This makes it a very appropriate file format for specifying rock physics models details about the constituents and their composition required when modelling the effective properties of the rock. To support this statement, it can be mentioned that the modelling results presented in this paper were generated using a software programme which read, parsed and automatically performed the specified modelling in each of the Rock.XML files.

In this paper, our focus has been on the philosophy instead of

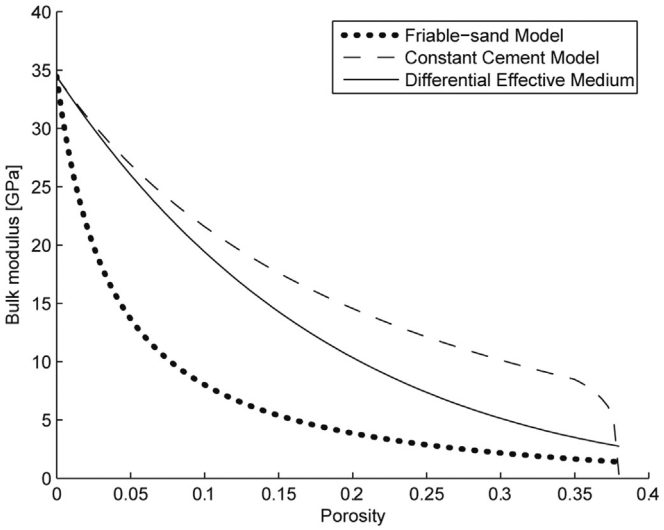

Fig. 5. Bulk modulus plotted versus porosity for model 1, 2 and 3.

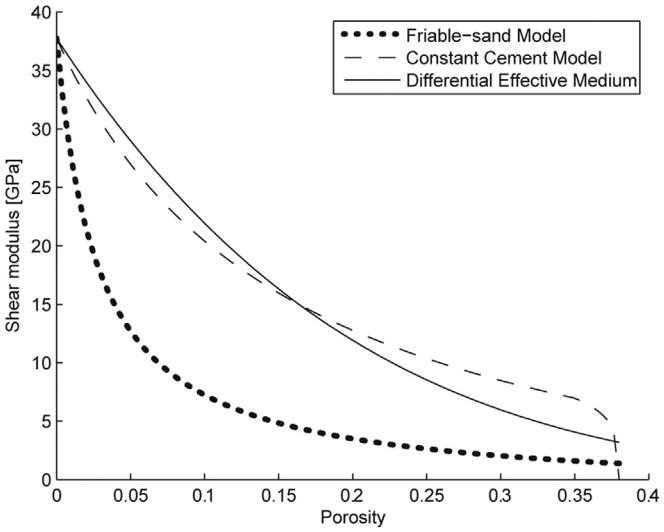

Fig. 6. Shear modulus plotted versus porosity for model 1, 2 and 3.

the actual implementation. Even though we provide a manual for the current version of Rock.XML, we do not claim our implementation of this philosophy is the best. But we consider this to be an important first step in the right direction, which makes it possible for others to start using the framework and referring to it using the Rock.XML file format. However, there is of course room for improvements. One can for example include units as attributes to XML elements. There is the possibility of using the XML Schemas (Ray, 2003) as a quality control to make sure the file format is according to the specifications of Rock.XML. Another possible improvement is to use namespaces (Harold and Means, 2004) to separate it from other XML code if integrated into a software package, or as identifiers for different parts within Rock.XML.

It is important to note that Rock.XML in itself does not provide the details or implementation of the various theories (functions). E.g. it only says one should use the theory of Voigt (1928) in a particular rock physics model to calculate the effective properties of a patchy fluid saturation. But it does not explain what the Voigt theory is or how it is implemented. Hence, if creating a parser to use a Rock.XML file to calculate the effective elastic properties of a rock, one would need to call on an already defined Voigt function to perform the actual calculation. One would have to create this function oneself, or find it in a function library, such as the downloadable library of MATLAB codes associated with the Rock Physics Handbook by Mavko et al. (2009).

We have only included three examples of isotropic rock physics models in this paper. But Rock.XML can in principle handle any model (isotropic or anisotropic) which exists today or in the future. If a model should become so complex or intricate making it basically impossible to decompose into building blocks as we have done here, or follow the general modelling strategy outlined in the introduction, then one can always implement it on a larger scale, i.e. just naming the model and all its required parameters. For example, the friable-sand model could have been implemented more as a black box where specifications of mineral and fluid properties, high porosity endmember, sorting and fluid effects were extracted out of the Rock.XML file and into a friable-sand model function. However, this can come at the expense of reusability, transparency and model customization.

5. Conclusions

We have presented a method for digitally storing specifications of a rock (its components and composition) for the purpose of modelling rock properties such as density, elastic moduli and acoustic velocities. This ensures the documentation and communication of models in a unique and unambiguous way which removes a source of error when doing modelling and simplifies the bookkeeping process. Our method is flexible and scalable; i.e. it can handle any current and future models. It can become a very useful component in rock physics modelling software by allowing storing specified models for later use, or import and exporting them between software. It opens up the possibility to create a library of rock physics models making them easily accessible to geoscientists around the world.

Acknowledgement

We would like to thank The Research Council of Norway (Petromaks and Climit programs; NFR project 207097/E20), Statoil and ExxonMobil for financial support.

Appendix A. Supplementary material

XML markup language and Rock.XML files can be downloaded from https://github.com/cageo/Jensen-2016.

References

Avseth, P., Dvorkin, J., Mavko, G., Rykkje, J., 2000. Rock physics diagnostic of North Sea sands: link between microstructure and seismic properties. Geophys. Res. Lett. 27, 2761–2764.

Avseth, P., Mukerji, T., Mavko, G., Dvorkin, J., 2010. Rock-physics diagnostics of depositional texture, diagenetic alterations, and reservoir heterogeneity in high-porosity siliciclastic sediments and rocks – A review of selected models and suggested work flows. Geophysics 75, 75A31–7547.

Avseth, P., Johansen, T., Bakhorji, A., Mustafa, H., 2014. Rock-physics modeling guided by depositional and burial history in low-to-intermediate-porosity sandstones. Geophysics 79, D115–D121.

Avseth, P., Skjei, N., Mavko, G., 2012. Rock Physics Modelling of Stress Sensitivity in Patchy Cemented Sandstones. In: Proceedings of the 74th EAGE Conference & Exhibition, Copenhagen.

Babaie, H.A., Babaei, A., 2005. Modeling geological objects with the XML Schema. Comput. Geosci. 31, 1135–1150.

Batzle, M., Wang, Z., 1992. Seismic properties of pore fluids. Geophysics 57, 1396–1408.

Berryman, J.G., 1992. Single-scattering approximations for coefficients in Biot's equations of poroelasticity. J. Acoust. Soc. Am. 91, 551–571.
Berryman, J.G., 1995. Mixture theories for rock properties. In: Ahrens, T.J. (Ed.), Physics and Phase Relations, A Handbook of Physical Constants. American Geophysical Union, Washington, D.C., pp. 205–228.
Dvorkin, J., Nur, A., 1996. Elasticity of high-porosity sandstones: theory for two North Sea data sets. Geophysics 61, 1363–1370.
Dvorkin, J., Mavko, G., Nur, A., 1991. The effect of cementation on the elastic properties of granular material. Mech. Mater. 12, 207–217.
Dvorkin, J., Nur, A., Yin, H., 1994. Effective properties of cemented granular materials. Mech. Mater. 18, 351–366.
Gassmann, F., 1951. Über die elastizität poröser medien. Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich 96, 1–23.
Harold, E.R., Means, W.S., 2004. XML in a Nutshell. O'Reilly Media, Sebastopol, CA.
Houlding, S.W., 2001. XML — an opportunity for < meaningful > data standards in the geosciences. Comput. Geosci. 27, 839–849.
Johansen, T., Jensen, E., Mavko, G., Dvorkin, J., 2013. Inverse rock physics modeling for reservoir quality prediction. Geophysics 78, M1–M18.
Kolbjørnsen, O., Ulvmoen, M., Hauge, V. L., Hauge, R., Fjellvoll, B., Jensen, E. H., Johansen, T. A., Dvorkin, J., Mavko, G., Furre, A.-K., Hohnston, D., 2013. Monitoring Geological CO_2 Storage, Poster at CLIMIT SUMMIT 2013; Oslo. Available at: ⟨http://www.nr.no/en/nrpublication?query=/file/1403166290/MonCO2_Workflow_Kolbjornsen_2013.pdf⟩, 25.02.2013.
Lake, R., 2005. The application of geography markup language (GML) to the geological sciences. Comput. Geosci. 31, 1081–1094.
Li, X., Wang, G., Zhu, H., 2012. A data model for exchanging and sharing ISRM rock mechanics test. Electron. J. Geotechn. Eng. 17, 25.
Mavko, G., Mukerji, T., Dvorkin, J., 2009. The Rock Physics Handbook: Tools for Seismic Analysis of Porous Media. Cambridge University Press, Cambridge.
Mello, U.T., Xu, L., 2006. Using XML to improve the productivity and robustness in application development in geosciences. Comput. Geosci. 32, 1646–1653.
Nance, K.L., Hay, B., 2005. Automatic transformations between geoscience standards using XML. Comput. Geosci. 31, 1165–1174.
Nativi, S., Caron, J., Davis, E., Domenico, B., 2005. Design and implementation of netCDF markup language (NcML) and its GML-based extension (NcML-GML). Comput. Geosci. 31, 1104–1118.
Ray, E.T., 2003. Learning XML. O'Reilly Media, Sebastopol, CA.
Sen, M., Duffy, T., 2005. GeoSciML: Development of a generic GeoScience Markup Language. Comput. Geosci. 31, 1095–1103.
Victorine, J., Watney, W.L., Bhattacharya, S., 2005. Use of XML and Java for collaborative petroleum reservoir modeling on the internet. Comput. Geosci. 31, 1151–1164.
Voigt, W., 1928. Lehrbuch der Kristallphysik (mit Ausschluss der Kristalloptik). B.G. Teubner J.W. Edwards, Leipzing und Berlin; Ann Arbor, Mich.
Walpole, L.J., 1966a. On bounds for the overall elastic moduli of inhomogeneous systems – I. J. Mech. Phys. Solids 14, 151–162.
Walpole, L.J., 1966b. On bounds for the overall elastic moduli of inhomogeneous systems – II. J. Mech. Phys. Solids 14, 289–301.
Walton, K., 1987. The effective elastic moduli of a random packing of spheres. J. Mech. Phys. Solids 35, 213–226.