xPand: An algorithm for perturbing homogeneous cosmologies

Cyril Pitrou[1], Xavier Roy[2] and Obinna Umeh[2]
[1] *Institute of Cosmology & Gravitation, University of Portsmouth, Portsmouth PO1 3FX, United Kingdom*
[2] *Department of Mathematics and Applied Mathematics, Cape Town University, Rondebosch 7701, South Africa*
(Dated: December 31, 2012)

In the present paper, we develop in details a fully geometrical method for deriving perturbation equations about a spatially homogeneous background. This method relies on the $3+1$ splitting of the background space-time and does not use any particular set of coordinates: it is implemented in terms of geometrical quantities only, using the tensor algebra package *xAct* with its extension for perturbations *xPert*. Our algorithm allows for the obtention of the perturbation evolution equations for all types of homogeneous cosmologies, up to any order and in all possible gauges. As applications, we checked that we recover the well-known perturbed Einstein equations in the Newtonian gauge, for Friedmann-Lemaître (FL) cosmologies up to second order and for Bianchi I cosmologies at first order. With our method, these equations are now easily found in any gauge and possibly for higher orders. This work also opens the door to the study of other perturbed Bianchi models, by circumventing the usually too cumbersome derivation of the perturbed equations, the major difficulty remaining the mode decomposition of perturbations on the homogeneous hypersurfaces. The main functions of the algorithm are implement in a package named *xPand* which can be loaded as an extension of *xPert*, enjoying thus the full power of *xAct* for tensor manipulations.

PACS numbers: 98.80.-k

INTRODUCTION

Cosmological perturbation theory is the cornerstone of our current understanding of origin and evolution of structures we see on the night sky today. Its role in cosmology dates back to the inflationary epoch, where the seed of structure formation resulted from the instability in the false vacuum of the inflaton field. We interpret this within perturbation theory as tiny quantum fluctuations on a de Sitter or quasi-de Sitter spacetime. The tiny quantum fluctuations become classical perturbations when their wavelength is stretched beyond the horizon. As soon as the classical perturbations re-enter the horizon, their physical evolution from the time of re-entry till now is understood within cosmological perturbation theory. This picture of the universe have been confirmed by the WMAP experiment [1]. Within the past decade many land and space based cosmological experiment such as the EUCLID and SKA have also been planned and they have the potential to transform cosmology into a precision subject. This dawn of precision cosmology is going to require our theoretical understanding of perturbation theory way beyond the linear order and the key question is: Are we theoretically ready?

Cosmological perturbation theory at linear order is simple and straight-forward to understand but it is grossly inadequate for understanding the late-time evolution of the universe where self-gravity of large scale structures and many other non-linear effects associated with gravitational collapse may carry the key Physics of interest. Going beyond first order in perturbation theory is a very difficult task, in fact it is almost impossible to perform even a coordinate or gauge transformation and at the same time be able keep track of every term by hand. To the best of our knowledge, there is no available easy-to-use software out there, that is solely designed to calculate all equations of motion for all the cosmological perturbation variables needed for interpreting cosmological observations in the precision cosmology era. The only closely available option is the GRTensor [2] which runs on Maple or Mathematica, however, the output of GRTensor at linear order is already very complicated to understand, let alone its output at non-linear order, the feature has made its used very unattractive.

To fill up this crucial missing gap in Theoretical Cosmology, we have developed a cosmological perturbation theory algebra package called *xPand*, which uses the tools of a tensor algebra package 'xAct/*xPert*' [3] to derive all the necessary equations for cosmological perturbation variables at any order in perturbation theory and for any dimension of spacetime. The xAct/xPert package runs on Mathematica and could be downloaded for free here [4]. The xAct/xPert package was specifically designed to handle perturbations on arbitrary background [3, 5]. It has already been applied to a spherically symmetric background space-time, more precisely to a Schwartzschild solution of the Einstein field equation [6, 7].

The interaction between *xPand* and xAct/*xPert* that we exploit here is such that we use the latter to expand the metric and all the curvature fields as perturbations around a homogeneous background space-time up to any order in perturbation theory. Afterwards the perturbation of the metric which now live on the background spacetime are decomposed with respect to the ADM 4-vector [8]. Base on this decomposition, we can relate the result to the standard Scalar-Vector-Tensor (SVT) decomposition of the perturbed metric. At present, *xPand* can handle perturbation around the Minkowski background spacetime, FL spacetime of different spatial curvature, Bianchi I and Bianchi A spacetimes. It is designed in such away that the user retains the freedom to choose the the background spacetime, the gauge, the indices, and formating or printing of the resulting equations. *xPand* offer the following gauge choices gauge choices : Any gauge, Comoving gauge, Flat gauge, Iso-density gauge, Newtonian gauge and Synchronous gauge. It is also possible for the user to define his/her gauge apart from the ones currently covered.

This paper is organized as follows: in section I, we provide a general overview of the mathematical framework on which *xPert* is built, This section is further split into three; we discuss in the first subsections the formalism for perturbing the metric, the second subsection deals with perturbation of curvature tensors, while the third subsection provides a comprehensive treatment of conformal transformation. In section II, we foliate the background manifold into parts parallel to 4-vector n^μ and the hypersurface orthogonal to the n^μ. While in section III, we decompose the perturbed metric with respect to n^μ, from where we define scalar, vector and tensor perturbations. The decomposition of the background covariant derivative and Lie derivative acting on scalar, vector and tensor perturbations in $1+3$ formalism is presented in section IV. In section V, we summarize the key aspects of the package and discuss the results in section VI. The final conclusion is presented in section VI B.

I. PERTURBATIONS AROUND A GENERAL SPACE-TIME

A. General framework

In this section we briefly review the algorithm of *xPert*, which constitutes the basis of our method. For more details, we refer the reader to, e.g., [3, 9, 10].

Let $\bar{\mathcal{M}}$ be the background manifold and $\mathcal{M}$ the perturbed (physical) manifold. Both can be related by means of a diffeomorphism $\phi : \bar{\mathcal{M}} \to \mathcal{M}$; tensorial quantities are thus transported from one manifold to the other with the help of the associated pull-back $\phi^\star$ and push-forward $\phi_\star$, along with their inverses. The metric of the perturbed space-time relates to that of the background as

$$\phi^\star(\boldsymbol{g}) \equiv \bar{\boldsymbol{g}} + \Delta[\bar{\boldsymbol{g}}] \equiv \bar{\boldsymbol{g}} + \sum_{n=1}^{\infty} \frac{\Delta^n[\bar{\boldsymbol{g}}]}{n!} \,. \tag{1}$$

It is convenient to note the $n^{\rm th}$ order perturbation of the metric as ${}^{\{n\}}\boldsymbol{h} \equiv \Delta^n[\bar{\boldsymbol{g}}]$. Here and in the sequel, we use boldface symbols for tensorial quantities, an over-bar for background quantities and a left superscript to denote the order of the perturbations. Unless otherwise specified, when we write down the components of a tensor, those should be understood as expressed in a general (arbitrary) basis (this holds equally for the background and perturbed tensors, and for the perturbations). Since all perturbation orders live on the background space-time as they are the result of the pullback of a physical tensorial quantity living in the perturbed manifold, we shall also raise and lower indices using the background metric; e.g. we have:

$${}^{\{n\}}h^{\mu\nu} \equiv {}^{\{n\}}h_{\alpha\beta}\, \bar{g}^{\mu\alpha}\, \bar{g}^{\nu\beta} \,, \tag{2}$$

for the $n^{\rm th}$ order of the metric perturbation. The definition of the perturbations clearly depends on the diffeomorphism ϕ, that is on the gauge choice, and in principle one should write Δ_ϕ instead of Δ. However, we shall omit any reference to the diffeomorphisme, that is to the gauge, and we use for the sake of clarity the notation: $\boldsymbol{T} \equiv \phi^\star(\boldsymbol{T})$, for any perturbed quantity. We will also use throughout this paper the notation $\Delta[\boldsymbol{T}]$ and $\Delta^n[\boldsymbol{T}]$ for the total pertrbation and the $n^{\rm th}$ order perturbation of a given tensor $\boldsymbol{T}$, following the notation of Ref. [3].

B. Expansion of the curvature tensors

The inverse of the metric tensor is obtained from the relation:

$$\boldsymbol{g}^{-1} = \left(\bar{\boldsymbol{g}} + \Delta[\bar{\boldsymbol{g}}]\right)^{-1} \,, \tag{3}$$

and it can be expanded as

$$\boldsymbol{g}^{-1} = \bar{\boldsymbol{g}}^{-1} \sum_{m=0}^{\infty} (-1)^m \left(\bar{\boldsymbol{g}}^{-1} \Delta[\bar{\boldsymbol{g}}]\right)^m \,, \tag{4}$$

and we thus obtain the $n^{\rm th}$ order perturbation of the inverse of the metric as

$$\Delta^n \left[\left(g^{-1}\right)^{\mu\nu}\right] = \sum_{(k_i)} (-1)^m \frac{n!}{k_1! \ldots k_m!}\; {}^{\{k_m\}}h^{\mu\lambda_m}\; {}^{\{k_{m-1}\}}h_{\lambda_m}{}^{\lambda_{m-1}} \ldots {}^{\{k_2\}}h_{\lambda_3}{}^{\lambda_2}\; {}^{\{k_1\}}h_{\lambda_2}{}^{\nu} \,. \tag{5}$$

The sum $\sum_{(k_i)}$ runs over the 2^{n-1} sorted partitions of n for $m \le n$ positive integers, such that $k_1 + \ldots k_m = n$. Note, importantly, that: $\Delta^n[(g^{-1})^{\mu\nu}] \neq {}^{\{n\}}h^{\mu\nu} \equiv {}^{\{n\}}h_{\alpha\beta}\, \bar{g}^{\mu\alpha}\, \bar{g}^{\nu\beta}$. For instance at first order $\Delta^1[(g^{-1})^{\mu\nu}] = -\,{}^{\{1\}}h^{\mu\nu}$

The difference between the Levi-Civita connection $\boldsymbol{\nabla}$ associated to the perturbed metric and the Levi-Civita connection $\bar{\boldsymbol{\nabla}}$ assocaited to the background metric is expressed by a rank $(1,2)$ tensor $\boldsymbol{\Gamma}$ such that for any form field $\boldsymbol{\omega}$

$$\nabla_\alpha \omega_\mu = \overline{\nabla}_\alpha \omega_\mu - \Gamma^\nu_{\alpha\mu}\omega_\nu. \tag{6}$$

It can then be shown that the perturbations of $\boldsymbol{\Gamma}$ are [3]

$$\Delta^n[\Gamma^\alpha_{\mu\nu}] = \sum_{(k_i)} (-1)^{m+1} \frac{n!}{k_1! \ldots k_m!}\; {}^{\{k_m\}}h^{\alpha\lambda_m}\; {}^{\{k_{m-1}\}}h_{\lambda_m}{}^{\lambda_{m-1}} \ldots {}^{\{k_2\}}h_{\lambda_3}{}^{\lambda_2}\; {}^{\{k_1\}}h_{\lambda_2\mu\nu} \,, \tag{7}$$

where we have defined the last term as

$${}^{\{n\}}h_{\alpha\mu\nu} \equiv \frac{1}{2}\left(\bar{\nabla}_\nu\, {}^{\{n\}}h_{\alpha\mu} + \bar{\nabla}_\mu\, {}^{\{n\}}h_{\alpha\nu} - \bar{\nabla}_\alpha\, {}^{\{n\}}h_{\mu\nu}\right) \,. \tag{8}$$

The perturbation of the Riemann tensor is given in all generality by

$$\Delta^n[R_{\mu\nu\alpha}{}^{\beta}] = \bar{\nabla}_\nu \left(\Delta^n[\Gamma^\beta_{\mu\alpha}]\right) - \sum_{k=1}^{n-1} \binom{n}{k} \Delta^k[\Gamma^\lambda_{\nu\alpha}]\, \Delta^{n-k}[\Gamma^\beta_{\lambda\mu}] \;-\; (\mu \leftrightarrow \nu)\,, \tag{9}$$

where $(\mu \leftrightarrow \nu)$ denotes the repetition of the preceding expression with indices μ and ν exchanged. Given a compatible metric, it can be recast using (7) into:

$$\begin{aligned}\Delta^n[R_{\mu\nu\alpha}{}^{\beta}] = \sum_{(k_i)} (-1)^m \frac{n!}{k_1!\ldots k_m!} \Big[& {}^{\{k_m\}}h^{\beta\lambda_m} \;\ldots\; {}^{\{k_2\}}h_{\lambda_3}{}^{\lambda_2}\, \bar{\nabla}_\mu\, {}^{\{k_1\}}h_{\lambda_2\alpha\nu} \\ & + \sum_{s=2}^{m} {}^{\{k_m\}}h^{\beta\lambda_m} \;\ldots\; {}^{\{k_{s+1}\}}h_{\lambda_{s+2}}{}^{\lambda_{s+1}}\; {}^{\{k_s\}}h_{\lambda_s\lambda_{s+1}\mu}\; {}^{\{k_{s-1}\}}h^{\lambda_s\lambda_{s-1}} \;\ldots\; {}^{\{k_2\}}h_{\lambda_3}{}^{\lambda_2}\; {}^{\{k_1\}}h_{\lambda_2\nu\alpha}\Big] \;-\; (\mu \leftrightarrow \nu)\,.\end{aligned} \tag{10}$$

The perturbation of the Ricci tensor is simply obtained by contracting the second and fourth indices of $\Delta^n[R_{\mu\nu\alpha}{}^{\beta}]$ in the previous expression, and the perturbation of the Ricci scalar, $R = g^{\alpha\beta}R_{\alpha\beta}$, reads:

$$\Delta^n[R] = \sum_{k=0}^{n} \binom{n}{k} \Delta^k[g^{\alpha\beta}]\;\; \Delta^{n-k}[R_{\alpha\beta}]\,. \tag{11}$$

Finally, we write the perturbed Einstein tensor:

$$\Delta^n[G_{\mu\nu}] = \Delta^n[R_{\mu\nu}] - \frac{1}{2} \sum_{k=0}^{n}\sum_{j=0}^{k} \frac{n!}{k!\,j!\,(n-j-k)!}\; {}^{\{j\}}h_{\mu\nu}\, \Delta^k[g^{\alpha\beta}]\, \Delta^{n-j-k}[R_{\alpha\beta}]\,.$$

It is not necessary to further simplify the two last formulas, as their current forms are already quite efficient to compute the n^{th} term of the respective perturbations. All the perturbative expansions of this section are implemented already in the package *xPert* [3]. We review here very briefly its main functions. The packages can be loaded by evaluating

In[1] := `<<xAct‘xPert‘`

(Version and copyright messages)

We first define a 4 dimensinal manifold `M` with abstract indiced `{b,c,i,j}`, and a metric `g` with negative determinant and associated Levi-Civita covariant derivative `CD`

In[2] := `DefManifold[M, 4,b,c,i,j];`

In[3] := `DefMetric[-1, g[-b, -c], CD, ";", "∇̄"];`

Many tensors associated with the metric are then automatically defined, such as the Riemann tensor or the Ricci tensor. The main feature of the *xAct* package, is that it represent down indices with a minus sign in front of the index, and no sign in front of an up index. For instance `g[-b, -c]` means g_{bc}. The perturbations `dg` of the metric `g` are defined using

In[4] := `DefMetricPerturbation[ g, dg, ε ]`

and this also identifies ε as the perturbative parameter of the expansions. It is now possible to evaluate the perturbation of tensors associated to the metric. For instance the perturbation up to first order of the Ricci tensor is found by evaluating

In[5] := `ExpandPerturbation@Perturbed[RicciScalarCD[], 1] //ContractMetric//ToCanonical`

Out[5] := $R[\bar{\nabla}] - \varepsilon\, dg^{1bc} R[\bar{\nabla}]_{bc} + \varepsilon\, \bar{\nabla}_c \bar{\nabla}_b dg^{1bc} - \varepsilon\, \bar{\nabla}_c \bar{\nabla}^c dg^{1b}{}_b$

In the internal notation, the perturbation of the metric has a label index which specifies the order. For instance the first order perturbation is internally stored as `dg[LI[1], -b, -c]`. The couple of functions `ExpandPerturbation@Perturbed[expr,n]` is used to perturb an expression `expr` up to order `n`. Then the function `ContractMetric[]` was used to remove the appearance of the background metric wherever possible, and the function `ToCanonical[]` is called to simplify as much as possible the result, gathering together the terms which are equal up to symmetries. More details can be found in Ref. [3].

C. Conformal transformation

In cosmology, it proves convenient to employ a conformally transformed metric so as to separate the effects of the background expansion from the evolution of perturbations. This type of transformation preserves the null structure of space-time, and hence its causal structure. Here, we define the conformal metric and its inverse respectively by

$$\widetilde{g}_{\mu\nu} = a^2 g_{\mu\nu}\,, \qquad (\widetilde{g}^{-1})^{\mu\nu} = a^{-2} g^{\mu\nu}\,, \tag{12}$$

with a being the scale factor of the background manifold. It is worth notifying that we do note define the components $\widetilde{g}^{\mu\nu}$ with the latter expression, as is customary. The reason for such a definition stems from the fact that, in the algorithm, the *same* metric is used throughout the computation to raise and lower the indices. In effect, we always raise and lower the indices with $g_{\mu\nu}$ and $g^{\mu\nu}$. Accordingly, we obtain here the unusual relations:

$$\widetilde{g}^{\mu\nu} = a^2 g^{\mu\nu} = a^4 (\widetilde{g}^{-1})^{\mu\nu}\,, \quad \widetilde{g}^{\mu}{}_{\nu} = a^2 \delta^{\mu}{}_{\nu} \quad \Rightarrow \quad \widetilde{g}^{\mu\alpha}\widetilde{g}_{\alpha\nu} = a^4 \delta^{\mu}{}_{\nu}\,. \tag{13}$$

A perturbative expansion similar to Eq. (1) is then performed so that on the background level $\overline{\widetilde{g}}_{\mu\nu} = a^2 \bar{g}_{\mu\nu}$ and at the perturbed level ${}^{\{n\}}\widetilde{h}_{\mu\nu} = a^2\, {}^{\{n\}}h_{\mu\nu}$. We choose conventionnaly to raise and lower indices with $\bar{g}_{\mu\nu}$ and $\bar{g}^{\mu\nu}$ so that

$${}^{\{n\}}\widetilde{h}^{\mu\nu} = \bar{g}^{\mu\alpha}\bar{g}^{\nu\beta}\, {}^{\{n\}}\widetilde{h}_{\alpha\beta} \tag{14}$$

The true physical metric is $\widetilde{\boldsymbol{g}}$ and the scale factor a is chosen so that there is no cosmic expansion in $\bar{g}_{\mu\nu}$ (this will be defined more precisely further).

We note $\widetilde{\boldsymbol{\nabla}}$ (respectively $\overline{\widetilde{\boldsymbol{\nabla}}}$) the Levi-Civita connection associated with $\widetilde{\boldsymbol{g}}$ (respectively $\overline{\widetilde{\boldsymbol{g}}}$). Note that since by definition the scale factor of the conformal transformation a is not perturbed (that is $a = \bar{a}$), then $\overline{\widetilde{\boldsymbol{g}}} = \widetilde{\overline{\boldsymbol{g}}}$ so that $\overline{\widetilde{\boldsymbol{\nabla}}} = \widetilde{\overline{\boldsymbol{\nabla}}}$. We relate the two sets of connections via the relation:

$$\widetilde{\nabla}_{\mu}\omega_{\nu} = \nabla_{\mu}\omega_{\nu} - C^{\alpha}{}_{\mu\nu}\omega_{\alpha}\,, \qquad \overline{\widetilde{\nabla}}_{\mu}\omega_{\nu} = \overline{\nabla}_{\mu}\omega_{\nu} - \overline{C}^{\alpha}{}_{\mu\nu}\omega_{\alpha}\,, \tag{15}$$

for any one-form $\boldsymbol{\omega}$. The 'nabla-connectors' $C^{\alpha}{}_{\mu\nu}$, derived from the definitions (12), take the form [11]:

$$C^{\alpha}{}_{\mu\nu} \equiv 2\delta^{\alpha}{}_{(\mu}\nabla_{\nu)}\ln a - g_{\mu\nu}\nabla^{\alpha}\ln a\,, \quad \overline{C}^{\alpha}{}_{\mu\nu} \equiv 2\delta^{\alpha}{}_{(\mu}\overline{\nabla}_{\nu)}\ln a - \bar{g}_{\mu\nu}\overline{\nabla}^{\alpha}\ln a \tag{16}$$

where the parentheses imply symmetrization over the indices enclosed. Let us stress for clarity that $\nabla^{\alpha} \equiv g^{\alpha\beta}\nabla_{\beta}$ amd $\overline{\nabla}^{\alpha} \equiv \bar{g}^{\alpha\beta}\overline{\nabla}_{\beta}$, and since a is a scalar function, the covariant derivatives reduce to partial derivatives in the above expressions The perturbations of $C^{\alpha}{}_{\mu\nu}$ can be readily obtained, and we have:

$$\overline{C}^{\alpha}{}_{\mu\nu} = 2\delta^{\alpha}{}_{(\mu}\partial_{\nu)}\ln a - \bar{g}_{\mu\nu}\bar{g}^{\beta\alpha}\partial_{\beta}\ln a\,, \tag{17}$$

$$\Delta[C^{\alpha}{}_{\mu\nu}] = \sum_{k=0}^{n}\frac{n!}{k!(n-k)!}\,{}^{\{k\}}g_{\mu\nu}\;{}^{\{n\,-\,k\}}g^{\alpha\beta}\,\partial_{\beta}\ln a\,. \tag{18}$$

Note that the conformal transformation $\widetilde{\Gamma}^{\alpha}_{\mu\nu}$ of $\Gamma^{\alpha}_{\mu\nu}$ which is obtained by replacing ${}^{\{n\}}h_{\mu\nu}$ with ${}^{\{n\}}\widetilde{h}_{\mu\nu}$ in its expansion (7) satisfies

$$\widetilde{\Gamma}^{\alpha}_{\mu\nu} - \Gamma^{\alpha}_{\mu\nu} = \Delta[C^{\alpha}{}_{\mu\nu}] \quad \Rightarrow \quad \widetilde{\nabla_{\alpha}\omega_{\mu}} = \overline{\widetilde{\nabla}}_{\alpha}\widetilde{\omega_{\mu}} - \widetilde{\Gamma}^{\nu}_{\alpha\mu}\widetilde{\omega}_{\nu} = \overline{\nabla}_{\alpha}\widetilde{\omega_{\mu}} - \Gamma^{\nu}_{\alpha\mu}\widetilde{\omega}_{\nu} - C^{\nu}_{\alpha\mu}\widetilde{\omega}_{\nu} \tag{19}$$

meaning that it is equivalent to perform the transformations $\overline{\boldsymbol{\nabla}} \to \boldsymbol{\nabla}$ and $\boldsymbol{\nabla} \to \widetilde{\boldsymbol{\nabla}}$ or the transformations $\overline{\boldsymbol{\nabla}} \to \widetilde{\overline{\boldsymbol{\nabla}}}$ and $\widetilde{\overline{\boldsymbol{\nabla}}} = \overline{\widetilde{\boldsymbol{\nabla}}} \to \widetilde{\boldsymbol{\nabla}}$. It proves faster to use the latter, that is to perform first a conformal transformation, and then to perturb the result.

For instance the Riemann tensors of the metrics $\overline{\widetilde{g}}_{\mu\nu}$ and $\bar{g}_{\mu\nu}$ are related through the $\overline{C}^{\alpha}{}_{\mu\nu}$ as

$$\begin{aligned}
\overline{\widetilde{R}}_{\mu\nu\alpha}{}^{\beta} &= \overline{R}_{\mu\nu\alpha}{}^{\beta} - 2\overline{\nabla}_{[\mu}\overline{C}^{\beta}{}_{\nu]\alpha} + 2\overline{C}^{\sigma}{}_{\alpha[\mu}\overline{C}^{\beta}{}_{\nu]\sigma} \\
&= \overline{R}_{\mu\nu\alpha}{}^{\beta} + 2\delta^{\beta}{}_{[\mu}\overline{\nabla}_{\nu]}\overline{\nabla}_{\alpha}\ln a - 2\bar{g}_{\alpha[\mu}\overline{\nabla}_{\nu]}\overline{\nabla}^{\beta}\ln a \\
&\qquad - 2\delta^{\beta}{}_{[\mu}\overline{\nabla}_{\nu]}\ln a\overline{\nabla}_{\alpha}\ln a + 2\bar{g}_{\alpha[\mu}\overline{\nabla}_{\nu]}\ln a\overline{\nabla}^{\beta}\ln a - 2\bar{g}_{\alpha[\mu}\delta^{\beta}{}_{\nu]}\overline{\nabla}^{\sigma}\ln a\overline{\nabla}_{\sigma}\ln a\,,
\end{aligned} \tag{20}$$

where the brackets indicate anti-symmetrization over the indices enclosed. By perturbing this relation, we can then deduce how the relate the perturbation of the Rieman tensor associated to $\widetilde{g}_{\mu\nu}$, that is $\Delta^n\left[\overline{\widetilde{R}}_{\mu\nu\alpha}{}^{\beta}\right]$ to the perturbations of the Riemann tensor associated with the metric $g_{\mu\nu}$, that is $\Delta^n\left[\overline{R}_{\mu\nu\alpha}{}^{\beta}\right]$.

The *xAct* package provides the tools to define a metric conformally related to another, thanks to the option `ConformalTo` of the function `DefMetric`. We have encapsulated this in *xPert* in the function `DefConformalMetric`. This function also ensures the transitivity of the conformal transformation in case several conformally related metrics are already defined. We first load the package *xPand*

In[6] := `<<xAct‘xPand‘`

Package xPand‘ version 0.0.1, {2012,12,18}

CopyRight (C) 2012-2013, Cyril Pitrou, Xavier Roy and Obinna Umeh under the GPL.

Then the evaluation of

In[7] := `DefConformalMetric[g, a];`

defines a scale factor `a[]` and metric named `ga2` such that it is conformally related to `g` with scale factor `a`. We can then perform a conformal transformation by using the function `Conformal`. For instance

In[8] := `Conformal[g, ga2][ RicciScalarCD[] ]`

Out[8] := $\frac{R[\overline{\nabla}]}{a^2} - \frac{6\overline{\nabla}_b\overline{\nabla}^b a}{a^2}$

The conformal transformation can of course be performed on quantities which are not directly related to the Riemann tensor. For a general tensor, the conformal transformation is

$$\widetilde{T}^{\mu_1\dots\mu_q}_{\mu_1\dots\mu_p} = a^{p-q+W(T)}T^{\mu_1\dots\mu_q}_{\mu_1\dots\mu_p} \tag{21}$$

where $W(T)$ is the conformal weight of tensor. The default weight is 0 such that the norm of a given tensor is conserved in the conformal transformation but this can modified for each tensor by the user.

In[9] := `DefTensor[T[-b], M];`

In[10] := `Conformal[g,ga2][CD[-b]@T[-c]]`

Out[10] := $a\overline{\nabla}_b T_c + \overline{g}_{bc}T^i\overline{\nabla}_i a - T_b\overline{\nabla}_c a$

Up to here, after application of `Conformal` and `ExpandPerturbation@Perturbed` on an expression, we obtain its perturbed form, but expressed formally in function of the background tensors, the scale factor, and the background covariant derivative $\overline{\boldsymbol{\nabla}}$. In order to obtain differential equations, we shall use the homogeneity assumption of the spatial sections of the background space-time to perform a $3+1$ splitting on the background. This is presented in the next section.

II. $3+1$ SPLITTING OF THE BACKGROUND SPACE-TIME

A. Induced Metric

The assumption that the background space-time possesses a set of (three-dimensional) homogeneous surfaces provides a natural choice for the $3+1$ slicing. We foliate the background manifold by means of this family, and we denote by $\bar{\mathbf{n}}$ the unit time-like vector ($\bar{n}^\mu\bar{n}_\mu = -1$) normal to it. The metric of $\bar{\mathcal{M}}$ is decomposed accordingly as

$$\bar{g}_{\mu\nu} \equiv \bar{h}_{\mu\nu} - \bar{n}_\mu\bar{n}_\nu\,, \qquad \text{with} \qquad \bar{h}_{\mu\nu}\bar{n}^\mu = 0 \qquad \text{and} \qquad \bar{h}^\mu{}_\alpha\bar{h}^\alpha{}_\nu = \bar{h}^\mu{}_\nu\,, \tag{22}$$

where $\bar{\mathbf{h}}$ is the induced metric on the spatial hypersurfaces, and the acceleration of the so-called Eulerian observers satisfies [12]

$$\bar{a}_\mu \equiv \bar{n}^\alpha \,\bar{\nabla}_\alpha \bar{n}_\mu = \frac{\bar{D}_\mu \bar{\alpha}}{\bar{\alpha}}\,, \tag{23}$$

with $\bar{\alpha}$ being the lapse function. $\bar{\mathbf{D}}$ is the connection of the three-surfaces associated to $\bar{\mathbf{h}}$ ($\bar{D}_\alpha \bar{h}_{\mu\nu} = 0$), and it is related to the four-covariant derivative as

$$\bar{D}_\alpha T_{\mu_1 \ldots \mu_p} = \bar{h}^\beta{}_\alpha \bar{h}^{\nu_1}{}_{\mu_1} \ldots \bar{h}^{\nu_p}{}_{\mu_p} \bar{\nabla}_\beta T_{\nu_1 \ldots \nu_p}\,, \tag{24}$$

for any projected tensor field[1]. Since the lapse is homogeneous in the configuration at stake, the acceleration vanishes ($\bar{a}_\mu = 0$) and the observers are in geodesic motion. We can therefore label each hypersurface by its proper time η and write: $\bar{n}_\mu = -\bar{\nabla}_\mu \eta$. Also, $\bar{\mathbf{n}}$ being hypersurface-forming, its vorticity has to vanish by construction. This last property yields:

$$\bar{\omega}_{\mu\nu} \equiv \bar{h}^\alpha{}_\mu \bar{h}^\beta{}_{\bar{\nu}} \bar{\nabla}_{[\alpha} \bar{n}_{\beta]} = 0 \quad \Leftrightarrow \quad \bar{\nabla}_{[\mu} \bar{n}_{\nu]} = 0\,, \tag{25}$$

where the equivalence only holds thanks to the null acceleration. This background $3+1$ splitting can be understood as a particular case of the $1+3$ formalism (see Refs. [13, 14], or a particular case of the general $3+1$ formalism (see Ref. [12] for a review).

1. Extrinsic Curvature

Another tensor we shall make use of is the symmetric extrinsic curvature tensor, characterizing the way the three-surfaces are embedded into the background manifold. It satisfies the relation:

$$\bar{K}_{\mu\nu} = \bar{h}^\alpha{}_\mu \bar{h}^\beta{}_\nu \bar{\nabla}_\alpha \bar{n}_\beta\,, \tag{26}$$

where we have chosen a positive sign for the right-hand side[2]. From the decomposition (22) along with the vanishing of the acceleration $\bar{\mathbf{a}}$ and the unitarity of $\bar{\mathbf{n}}$, we can rewrite the previous expression as

$$\bar{K}_{\mu\nu} = \bar{\nabla}_\mu \bar{n}_\nu\,. \tag{27}$$

We choose the background metric $\bar{g}_{\mu\nu}$ ot be as free of expansion as possible, and to put the expansion in the scale factor of the conformal transformation. We are now in position to specify what we mean by this choice. In general we choose the extrinsic curvature to be traceless

$$\bar{K}_{\mu\nu} \bar{h}^{\mu\nu} = \bar{K}_\mu{}^\mu = 0 \tag{28}$$

Since $\boldsymbol{g}$ is not the true physical metric, it is always possible to satisfy this condition by choosing the scale factor which relates it to the physical metric $\widetilde{\boldsymbol{g}}$. The scale factor corresponds to the volume expansion, and the extrinsic curvature corresponds to the shear of the expansion so that we often use the notation $\bar{\sigma}_{\mu\nu} \equiv \bar{K}_{\mu\nu}$ with $\bar{\sigma}_\mu{}^\mu = 0$. In the case of maximally symmetric hypersurfaces (that is a FL space-time), since $\bar{K}_{\mu\nu} \propto \bar{h}_{\mu\nu}$, then this implies that the shear of expansion vanishes.

B. Gauss-Codazzi relations

Finally the splitting of the four-Riemann tensor can be constructed from its different projections onto the spatial slices and the congruence of the observers; it reads:

$$\bar{R}_{\mu\nu\alpha\beta} = {}^3\bar{R}_{\mu\nu\alpha\beta} + 2\bar{K}_{\mu[\alpha}\bar{K}_{\beta]\nu} - 4\left(\bar{D}_{[\mu}\bar{K}_{\nu][\alpha}\right)\bar{n}_{\beta]} - 4\left(\bar{D}_{[\alpha}\bar{K}_{\beta][\mu}\right)\bar{n}_{\nu]} + 4\,\bar{n}_{[\mu}\,\bar{K}_{\nu]}{}^\lambda\,\bar{K}_{\lambda[\alpha}\,\bar{n}_{\beta]} + 4\,\bar{n}_{[\mu}\,\dot{\bar{K}}_{\nu][\alpha}\bar{n}_{\beta]}\,, \tag{29}$$

[1] We recall that the operator $\bar{\mathbf{D}}$ looses its character of derivative when it is applied to non-spatial tensors. More precisely, we are not allowed to use the Leibniz rule anymore, as one can realize upon writing for instance:

$$\begin{aligned}\bar{D}_\alpha(\psi\,\bar{T}_{\mu_1\ldots\mu_p}) &= \psi\bar{D}_\alpha\bar{T}_{\mu_1\ldots\mu_p} + \bar{h}^{\nu_1}{}_{\mu_1}\ldots\bar{h}^{\nu_p}{}_{\mu_p}\bar{T}_{\nu_1\ldots\nu_p}\bar{D}_\alpha\psi\\ &\neq \psi\bar{D}_\alpha\bar{T}_{\mu_1\ldots\mu_p} + \bar{T}_{\mu_1\ldots\mu_p}\,\bar{D}_\alpha\psi\,,\end{aligned}$$

for any scalar field ψ. One then has to make sure that the correct expression is used in such a situation.

[2] This convention does not affect the $3+1$ Einstein equations as written in terms of the kinematical quantities of the observers.

where ${}^3\bar{R}_{\mu\nu\alpha\beta}$ denotes the (three-)Riemann curvature of the hypersurfaces. The purely spatial projection of this expression only calls upon the two first terms, and it drives the Gauss–Codazzi relation:

$$h^{\rho}{}_{\mu}h^{\sigma}{}_{\nu}h^{\lambda}{}_{\alpha}h^{\chi}{}_{\beta}\,\bar{R}_{\rho\sigma\lambda\chi} = {}^3\bar{R}_{\mu\nu\alpha\beta} + 2\bar{K}_{\mu[\alpha}\bar{K}_{\beta]\nu}\,. \tag{30}$$

The one-time and three-space projection gives, from the next two terms, the Gauss–Mainardi relation that we have implicitly used in the derivation of the above commutation rules, and the last non-null projection (two-time and two-space) provides, from the last two terms, an evolution equation for the extrinsic curvature.
This decomposition can be performed using the function `GaussCodazzi` which is already included in *xAct*.

C. Curvature of the spatial sections

For a maximally symmetric hypersurfaces, that is a FL space-time, the Riemann tensor of the induced metric takes the form

$${}^{(3)}R_{\mu\nu\alpha\beta} = 2Kh_{\alpha[\mu}h_{\nu]\beta}, \quad {}^{(3)}R_{\mu\nu} = 2Kh_{\mu\nu}, \quad {}^{(3)}R = 6K\,. \tag{31}$$

However general homogeneous hypersurfaces are not necessarily maximally symmetric and these are classified according to their Bianchi type. This is detailed in appendix A.

D. Implementation of the $3+1$ splitting

The function `BackgroundSlicing` gathers all definitions for the splitting. From the background metric `g`, it first defines the normal vector `n`, and the induced metric `h`. Then it also defines a scale factor `ah` and the conformally related metric `gah2`. The type of background space-time is also given as an argument to `BackgroundSlicing`. This background space-time is determined by the specification of the extrinsic curvature $\bar{K}_{\mu\nu}$ (it vanishes for FL space-times and is non-zero for general Bianchi space-times) and by the Riemann tensor of the induced metric. In case of maximally symmetric hypersurfaces, this is given by Eq. (31). For general Bianchi cases the function `BackgroundSlicing` also specifies the constants of structure of the spatial hypersurfaces. This is detailed as well in appendix A. For instance we can define the $3+1$ splitting of a curved FL space-time by evaluating

In[11] := `BackgroundSlicing[h, n, g, cd, {"|","D"}, FLCurved]`

Now, in order to go on step further in the $3+1$ splitting of perturbation, we remark that the induced derivative $\bar{\boldsymbol{D}}$ associated to $\bar{\boldsymbol{h}}$ can only act on tensors which are projected (that is which are invariant by projection with $\bar{h}_{\mu}^{\nu}$), so it is necessary to decompose first the perturbed tensors, among which the metric perturbations, into its components along $\boldsymbol{n}$ and its projected components along $\boldsymbol{n}$. The purpose of the next section is to review how this is performed.

III. PARAMETERIZATION OF THE PERTURBED TENSORS

A. Projected components and SVT decomposition

In general a tensor can be decomposed into its components along to $\boldsymbol{n}$ and along the projector $\bar{\boldsymbol{h}}$. For instance for the perturbation fo the metric, we have

$${}^{\{n\}}h_{\mu\nu} = \bar{n}_{\mu}\bar{n}_{\nu}\left({}^{\{n\}}h_{\alpha\beta}\bar{n}^{\alpha}\bar{n}^{\beta}\right) + \bar{n}_{\mu}\,{}^{\{n\}}h_{\alpha\beta}\bar{n}^{\alpha}\bar{h}^{\beta}_{\nu} + \bar{n}_{\nu}\,{}^{\{n\}}h_{\alpha\beta}\bar{n}^{\beta}\bar{h}^{\alpha}_{\mu} + {}^{\{n\}}h_{\alpha\beta}\bar{h}^{\alpha}_{\mu}\bar{h}^{\beta}_{\nu} \tag{32}$$

The components $\left({}^{\{n\}}h_{\alpha\beta}\bar{n}^{\alpha}\bar{n}^{\beta}\right)$, ${}^{\{n\}}h_{\alpha\beta}\bar{n}^{\alpha}\bar{h}^{\beta}_{\nu}$ and ${}^{\{n\}}h_{\alpha\beta}\bar{h}^{\alpha}_{\mu}\bar{h}^{\beta}_{\nu}$ are then projected scalar, vector and tensors. It is then suited to use the standard SVT decomposition for these projecetd components.

Any projected vector U_{μ} can be split into its scalar part S and its vector part V_{μ} according to

$$U_{\mu} \equiv \bar{D}_{\mu}S + V_{\mu} \quad \text{with} \quad \bar{D}^{\mu}V_{\mu} = 0\,. \tag{33}$$

V_{μ} is necessarily projected as well. Similarly, any projected symmetric and traceless tensor $H_{\mu\nu}$ (that is such that $H^{\mu}{}_{\mu} = 0$) can be split into its scalar part S, its vector part V_{μ} and its tensor part $T_{\mu\nu}$ according to

$$H_{\mu\nu} = \left(\bar{D}_{\mu}\bar{D}_{\nu} - \frac{1}{3}\bar{h}_{\mu\nu}\bar{D}_{\alpha}\bar{D}^{\alpha}\right)S + \bar{D}_{(\mu}V_{\nu)} + T_{\mu\nu} \tag{34}$$

with $\bar{D}^\mu V_\mu = \bar{D}^\mu T_{\mu\nu} = T^\mu_{\ \mu} = 0$. V_μ and $T_{\mu\nu}$ are necessarily projected as well. We shall use such type of SVT decomposition to split the projected components of the metric perturbations, but also for the projected components of any tensor on which we want to perform a perturbative expansion.

B. Metric perturbation parameterization

In the case of metric perturbations the SVT decomposition of the projected components are usually given by

$$
\begin{aligned}
{}^{\{n\}}h_{\mu\nu}\bar{n}^\mu\bar{n}^\nu &\equiv -2\,{}^{\{n\}}\Phi && \text{(35a)}\\
{}^{\{n\}}h_{\mu\alpha}\bar{n}^\mu\bar{h}^\alpha_\nu &\equiv -\bar{D}_\nu\,{}^{\{n\}}B - {}^{\{n\}}B_\nu && \text{(35b)}\\
\frac{1}{2}\,{}^{\{n\}}h_{\beta\alpha}\bar{h}^\beta_\mu\bar{h}^\alpha_\nu &\equiv -\bar{h}_{\mu\nu}\,{}^{\{n\}}\Psi + \bar{D}_\mu\bar{D}_\nu\,{}^{\{n\}}E + \bar{D}_{(\mu}\,{}^{\{n\}}E_{\nu)} + {}^{\{n\}}E_{\mu\nu} && \text{(35c)}
\end{aligned}
$$

For specific gauge choices, some of the fields used in the decomposition are required to vanish (see Ref. [15] for a review), but here, this decomposition is the most general since it parameterizes the 10 degrees of freedom of the perturbation. 4 scalar degrees of freedom are encoded in ${}^{\{n\}}\Phi$, ${}^{\{n\}}\Psi$, ${}^{\{n\}}E$, and ${}^{\{n\}}B$, 4 vector degrees of freedom are encoded in ${}^{\{n\}}E_\mu$ and ${}^{\{n\}}B_\nu$, and 2 tensor degrees of freedom remain in ${}^{\{n\}}E_{\mu\nu}$.

IV. $3+1$ SPLITTING OF THE PERTURBED EXPRESSION

A. Properties of the projected tensors

In order to complete the $3+1$ splitting of perturbed tensors, it is necessary to express the background covariant derivative $\overline{\boldsymbol{\nabla}}$ in function of the background induced derivative $\bar{\boldsymbol{D}}$. For a general tensor, the relation between these two derivatives reads

$$
\bar{\nabla}_\alpha T_{\mu_1\ldots\mu_p} = -\bar{n}_\alpha \dot{T}_{\mu_1\ldots\mu_p} + \bar{D}_\alpha T_{\mu_1\ldots\mu_p} + \sum_{i=1}^{p} \bar{n}_{\mu_i}\,\bar{K}^\beta_{\ \ \alpha}\,T_{\mu_1\ldots\mu_{i-1}\beta\mu_{i+1}\ldots\mu_p}\,, \tag{36}
$$

where the over-dot stands for the covariant derivative along the world-lines of the observers (for any tensor field, we have : $\dot{T}_{\mu_1\ldots\mu_p} \equiv \bar{n}^\alpha\overline{\nabla}_\alpha T_{\mu_1\ldots\mu_p}$). In the case of a vanishing acceleration, the dot derivative of a spatial tensor is itself a spatial tensor.

It will prove much more useful to use the Lie derivative along the congruence of $\bar{n}$ rather than the dot derivative. The reason comes from the fact that, if we choose a coordinates system x^i on a given hypersurface, and we drag it on all hypersurfaces with $\bar{n}$ so as to extend it on the entire manifold and obtain a coordinate system (η, x^i), then the action of $\mathcal{L}_{\bar{\mathbf{n}}}$ on a tensor is equivalent to that of $\partial/\partial\eta$. Even though our formalism is purely geometric, it aims eventually at deriving partial differential equations in η for perturbations and this will be made possible in the final stage, once all derivatives are expressed in function of $\mathcal{L}_{\bar{\mathbf{n}}}$ or $\bar{\boldsymbol{D}}$. The relation between the Lie and the dot derivatives is easily derived; it is expressed in terms of the extrinsic curvature as

$$
\mathcal{L}_{\bar{n}} T_{\mu_1\ldots\mu_p} = \dot{T}_{\mu_1\ldots\mu_p} \ + \ T_{\alpha\mu_2\ldots\mu_p}\,\bar{K}^\alpha_{\ \ \mu_1} \ + \ \ldots \ + \ T_{\mu_1\ldots\mu_{p-1}\alpha}\,\bar{K}^\alpha_{\ \ \mu_p}\,. \tag{37}
$$

Note that for a spatial tensor, $\mathcal{L}_{\bar{n}} T_{\mu_1\ldots\mu_p}$ is also spatial. From this last expression, we can recast relation (36) into

$$
\bar{\nabla}_\alpha T_{\mu_1\ldots\mu_p} = -\,\bar{n}_\alpha\,\mathcal{L}_{\bar{n}} T_{\mu_1\ldots\mu_p} \ + \ \bar{D}_\alpha T_{\mu_1\ldots\mu_p} + \ 2\,\bar{n}_{(\alpha}\,\bar{K}_{\mu_1)}^{\ \ \ \beta}\,T_{\beta\mu_2\ldots\mu_p} + \ \ldots \ + \ 2\,\bar{n}_{(\alpha}\,\bar{K}_{\mu_p)}^{\ \ \ \beta}\,T_{\mu_1\ldots\mu_{p-1}\beta}\,. \tag{38}
$$

Finally, let us remark that the $\mathcal{L}_{\bar{n}}$ and $\bar{\boldsymbol{D}}$ derivatives do not commute in general. For spatial tensors (with indices down) the commutation reads:

$$
[\mathcal{L}_{\bar{n}}, \bar{D}_\mu]\bar{T}_{\nu_1\ldots\nu_p} = \sum_{i=1}^{p}\left(\bar{h}^{\alpha\beta}\bar{D}_\beta\bar{K}_{\mu\nu_i} - \bar{D}_\mu\bar{K}_{\nu_i}^{\ \ \alpha} - \bar{D}_{\nu_i}\bar{K}_\mu^{\ \ \alpha}\right)\bar{T}_{\nu_1\ldots\nu_{i-1}\alpha\nu_{i+1}\ldots\nu_p}\,. \tag{39}
$$

The function `DefProjectedTensor` which is implemented in *xPand* uses the function `DefTensor` of *xAct* in order to defined a tensor, and then it implements all the necessary rules to specify that this tensor is projected. Obviously it defines that it vanishes once contracted with $\bar{\boldsymbol{n}}$ and is invariant once projected with $\bar{\boldsymbol{h}}$, but most importantly it implements the rule (39) among less significant rules. The rule (38) is not implemented automatically at this stage so that the user can keep a control on the smplifications. We will see further where this takes place. Some options can be also specified in `DefProjectedTensor` to add the symmetric, traceless and transverse conditions, so that the tensor defined can be readily used for a SVT decomposition. By default this is the case, and the options need to be specified if these conditions should not be satisfied. For instance, we can define the two Bardeen potentials Φ and Ψ and the tensor degree of freedom associated to gravitational waves by evaluating

In[12] := `DefProjectedTensor[`ϕ`[], h]; DefProjectedTensor[`ψ`[], h];`

In[13] := `DefProjectedTensor[Et[-b, -c], h, PrintAs->"E"];`

It is then possible to define a rule for the components of the metric perturbation

In[14] := `RuleMetric = dg[LI[ord_], i_ , j_] :>`
`-2 n[i]n[j]` ϕ`[LI[ord]] -2`ψ`[LI[ord]] h[i, j] + 2 Et[LI[ord],i, j];`

A comment is in order about the use of label index `LI[ord]`. In *xPert*, a label index `LI[n]` was used to denote the order of the perturbation. For instance `dg[LI[1], -b, -c]` is the first order perturbation of the metric. In *xPand*, for projected tensors which were defined thanks to the function `DefProjectedTensor`, we use the same notation but we also add a second label index which denotes the number of $\mathcal{L}_{\bar{\boldsymbol{n}}}$ which is applied to the tensor. Since the meaning Lie derivative is dependent on the position of indices, we conventionnally choose this second label index to be the number of Lie derivatives when the tensor has only down indices.

To summarize the notation, `Et[LI[1], LI[0], -b, -c]` is the first order tensor which is used in the parameterization of the metric perturbation. Some internal rules are defined so that for instance `Et[LI[1], -b, -c]` is automatically converted to `Et[LI[1], LI[0], -b, -c]` and this explains why in the definition of the rule `RuleMetric` only one label index was mentioned in the tensors. We then have `LieD[`$\bar{n}$`][ Et[LI[1], LI[0], -b, -c] ] = Et[LI[1], LI[1], -b, -c]`. For instance

In[14] := `Et[LI[1], -b, -c]`

Out[14] := ${}^{(1)}E_{bc}$

Note that the first label index is displayed as a prepended exponent. Then taking one Lie derivative along $\bar{\boldsymbol{n}}$ increments the second label index. A prime notation is used for the display form of the second label index. For instance

In[15] := `LieD[ n[b] ][ % ]`

Out[15] := ${}^{(1)}E'_{bc}$

Finally note that since $\mathcal{L}_{\bar{\boldsymbol{n}}} g_{\mu\nu} = 2\bar{K}_{\mu\nu}$ and $\mathcal{L}_{\bar{\boldsymbol{n}}} g^{\mu\nu} = -2\bar{K}^{\mu\nu}$, then in general $\mathcal{L}_{\bar{\boldsymbol{n}}} {}^{(1)}E^{bc} = {}^{(1)}E^{bc\prime} - 2\bar{K}^{b}_{i}\,{}^{(1)}Et^{ic} - 2\bar{K}^{c}_{i}\,{}^{(1)}Et^{bi}$. However for maximally symmetric hypersurfaces, that is for FL cosmologies, which is the example that we follow in this presentation of the package, the extrinsic curvature vanishes so that the position of the indices for the definition of the second label index is not ambiguous at all.

B. Decomposition of the background covariant derivative

We now have all the necessary tools to split completely the result of a tensor whose perturbations we are interested in. For instance, if we want to perturb the Ricci scalar, we first perform a conformal transformation from `g` to `gah2` (but express the result in function of the metric `g`), then we perturb the result and we replace the metric perturbations by their SVT components using the rule `RuleMetric`

In[16] := `Conformal[g, gah2][ RicciScalarCD[] ]`

In[17] := `MyR = ExpandPerturbation@Perturbed[%, 1]`

In[18] := `(MyR/.RuleMetric) //ProjectorToMetric //GradNormalToExtrinsicK`
`// ContractMetric // ToCanonical`

The final result has been simplified first by using the *xAct* functions `ProjectorToMetric` which replaces $\bar{h}_{\mu\nu}$ by $\bar{g}_{\mu\nu} - \bar{n}_\mu \bar{n}_\mu$ and `GradNormalToExtrinsicK` which replaces $\overline{\nabla}_\mu \bar{n}_\nu$ by $\overline{K}_{\mu\nu}$ and then by using the couple of function `ContractMetric` and `ToCanonical` so as to simplify the result as much as possible.

Since the rule (38) was not evaluated so far, the result involves only background covariant derivatives ($\overline{\boldsymbol{\nabla}}$). In order to use (38) and split the background covariant derivative into induced derivatives and Lie Derivatives along $\bar{\boldsymbol{n}}$ we use the function `SplitPerturbations`. For instance, we evaluate

In[19] := `ah[]`^{2} `SplitPerturbations[%, h]`

Out[18] := $6\mathcal{H}^2 + 6\mathcal{H}' + 6\mathcal{K} + \varepsilon\left(-12\mathcal{H}^{2\ (1)}\phi - 12\mathcal{H}'^{\ (1)}\phi - 6\mathcal{H}^{\ (1)}\phi' - 18\mathcal{H}^{\ (1)}\psi' - 6^{\ (1)}\psi'' + 12^{\ (1)}\psi\mathcal{K}\right.$
$\left.-2D_bD^{b\ (1)}\phi + 4D_bD^{b\ (1)}\psi\right)$

This applies as well the Gauss-Codazzi decomposition, the commutation of Lie and induced derivative (Eq. 39) being performed automatically, and leads to the expected result. However this is rather inefficient for general gauges at higher order. Instead the rule `RuleMetric` can be replaced in an optimized manner by the function `SplitPerturbations`. It consists essentially in evaluating the rule (38) on the projected components of the metric *before* specifying the SVT decomposition with `RuleMetric`. This requires to give the rule `RuleMetric` to SplitPerturbations as an argument :

In[20] := `SplitPerturbations[MyR, RuleMetric, h]`

and takes much less time at higher orders.

V. SUMMARY OF THE ALGORITHM

A. Main steps

The main steps are

- i) We define the background manifold (`DefManifold`) with its associated metric (`DefMetric`) and then the background space-time type and the associated $3+1$ splitting is performed using `BackgroundSlicing`;
- ii) For the quantitiy we are interested to perturb, we first us its general expression in function of the expansionless metric $\boldsymbol{g}$ and then perform a conformal transformation with to express it in function of the physical metric $\widetilde{\boldsymbol{g}}$, but we rexpress the result in function of the metric $\boldsymbol{g}$. This is done with `Conformal`;
- iii) We use the basic tools of *xPert* (`Perturbed` and `ExpandPerturbation`) to perturb formally in functions of the perturbations of the metric and of the other tensors;
- iv) We define projected tensors with `DefTensorProjected` in order to define rules for the (SVT) decomposition of the projected components of tensor perturbations; It is possible at this stage to use the function `MatterSplitting` and `MetricSplitting` (see below) for the most common cases which correspond to the most common gauge choices and most common fields that need to be perturbed (metric, energy density, pressure, fluid velocity, scalar field). However, the user is free to use his own parameterization of the projected components of perturbations.
- v) We finally use these parameterization rules to replace them in an optimal way in the perturbed expression and split the background covariant derivative using `SplitPerturbations`. Gauss-Codazzi relations and commutation of Lie and induced derivatives are then automatically performed, and the results is simplified as much as possible.

The function `ToxPand` has been designed to perform nearly all these steps but the first at once, thus simplifying the computation of perturbations for the simplest cases.

B. A minimal example

We write here a brief example where the user has full control on the perturbation parameterization. In this case only the Bardeen potentials are kept in the metric perturbations:

```
<<xPand/xPand.m;
DefManifold[M, 4,{b,c,d,f}];
DefMetric[-1, g[-b, -c], CD, {";", "CD"}];
DefMetricPerturbation[g, dg, epsilon];
BackgroundSlicing[h, n, g, cd, {"|", "D"}, FLCurved];
order = 1;
DefProjectedTensor[Phi[], h];
DefProjectedTensor[Psi[], h];
```

```
MyRicciScalar=ExpandPerturbation@Perturbed[Conformal[g, gah2][RicciScalarCD[]], order];
RulesForMetric = {dg[LI[ord_], b_, c_] :> -2 n[b]n[c]Phi[LI[ord]] -2 h[b, c]Psi[LI[ord]]};
SplitPerturbations[MyRicciScalar, RulesForMetric, h]
```

If we want to use a predefined gauge for the metric perturbations, it is enough to use the function `ToxPand` which does nearly everything at once. For instance, these five lines are enough to obtain the perturbation of the Ricci tensor in any gauge up to second order.

```
<<xPand/xPand.m;
DefManifold[M, 4,{b,c,d,f}];
DefMetric[-1, g[-b, -c], CD, {";", "CD"}];
BackgroundSlicing[h, n, g, cd, {"|", "D"}, FLCurved];
ToxPand[RicciScalarCD[], dg, u, du, h, AnyGauge, 2]
```

C. Advanced functions for metric and matter perturbations

Even tough the user is free to parameterize the tensor perturbations (among which the metric perturbations), using rules with projected tensors defined with `DefProjectedTensor`, we provide a function `MetricSplitting` which creates the most standard rules for the usual gauge choices. For instance, the Newton gauge perturbation can be obtained by evaluation of

In[21] := `dg[LI[1], -b, -c] /.MetricSplitting[g, dg, h, NewtonGauge]`

Out[21] := $2\,{}^{(1)}E_{bc} - {}^{(1)}B_c n_b - {}^{(1)}B_b n_c - 2n_b n_c\,{}^{(1)}\phi - 2h_{bc}\,{}^{(1)}\psi$

Similarly, the function `MatterSplitting` creates the standard rules for usual gauges in order to replace the most common matter field perturbations. Among these, the less standard parameterization is the one of the fluid velocity, since the time component (that is the component along $\bar{\boldsymbol{n}}$) is determined from the projected part of the perturbation, thanks to the normalization condition. We also need to mention that since *xPert* provides also a function to perform order n general gauge transformations with the function `GaugeChange`, the function `SplitGaugeChange` of *xPand* extends this function for the $3+1$ splitting for the transformation rule of a tensor perturbation. For a glance at the gauge transformation of metric perturbation and the construction of gauge-invariant variables, we refer to Refs. [16–19].

For examples of use of the advanced functions `MetricSplitting`, `MatterSplitting` and `SplitGaugeChange`, the reader should consider going through the example notebooks which are distributed along with the package *xPand*. Finally, the function `ToxPand` which was presented in the section V B is a function which combines all the advanced functions so that the user can obtain the desired perturbations in a given gauge without having to deal with any detail of the algorithm.

VI. RESULTS

A. Recovering standard results

We have checked that with our implementation, we recover the standard results of perturbation theory in cosmology. More precisely, we recover

- all first order results of Einstein equation and stress-energy tensor conservation equation, for flat and curved FL space-times (see for instance [15]);
- all first order results for Bianchi type I background, in the gauge chosen in Ref. [20];
- second order perturbations of Einstein and stress-energy conservation equation, around flat and curved FL universe in the Newtonian gauge (see for instance [20] for the complete set of equations);

Our package now enables to extend these results to any gauge choice and to higher order if necessary.

It is worth stressing that, in order to obtain useful standard differential equations in the conformal time η, it is also necessary to perform a mode expansion on the hypersurfaces, that is to find the eigenmodes of the spatial Laplacian $\bar{D}_\mu \bar{D}^\mu$. This is easy for flat FL cosmologies where we can just use a Fourier transformation, and it is also well known for curved FL cosmologies where hyperspherical Bessel functions should be used [21]. However, it is still unknown for general Bianchi cosmologies. Apart form the special case of Bianchi type I, where the modes can be found from a Fourier transformation and thus lead to simple equations [22], there is no general technique to obtain the eigenmodes of the Laplacian for all other types of Bianchi spaces, and only in special cases (see for instance Ref. [23]) this has been done explicitely.

B. Timings

In practice, the timing grows like power law of the order of perturbations (see Fig. 1).

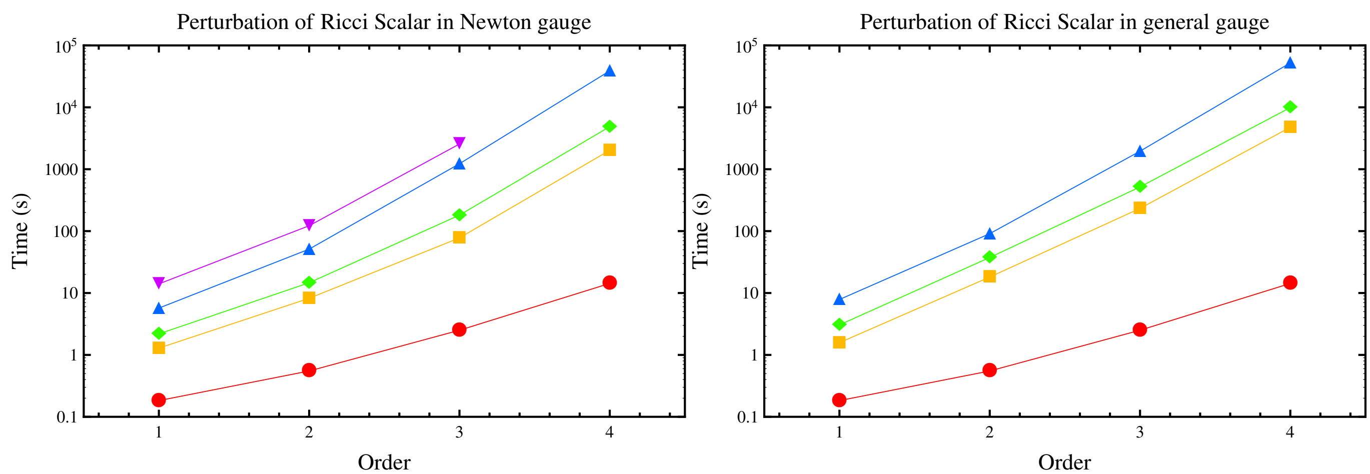

FIG. 1. Timing for the perturbation of the Ricci scalar. Left: restriction to Newtonian Gauge, and right: general gauge parameterization. From bottom to top on each plot: i) Formal perturbations with *xPert*and conformal transformation (in red); ii) Splitting around a Minkowski background (in yellow); iii) Splitting around a curved FL background (in green); iv) Splitting around a Bianchi I background (in blue); v) Splitting around a general Bianchi background (in purple).

TODO: Extend to gauge restrictions, Ricci scalar around curved FL, Ricci scalar around Bianchi I, Ricci scalar around Bianchi General... Who wants to do all this benchmarking on a given machine?

CONCLUSION

The package we have developed is the first comprehensive package that could perform algebraic perturbation theory calculation in cosmology up to any order of interest. It does this in a manner familiar to most cosmologists. The package avoids the complexities associated with component by component computation through the use of well understood $1+3$ decomposition formalism to split the perturbed variables on a given homogenous background spacetime and up to any order in perturbation theory. It is worth reiterating, the several advantages associated with the approach adopted in development of this outstanding package, those advantages include:

- the freedom to choose a style of formatting/printing of any declared variable by the user.
- the package handles indices just the same way the user would handle them if he/she were to do the calculation by hand. It totally eliminates the laborious summation of repeated indices can have inhibited the use other packages developed to solve a similar problem.
- It is easy to extend any computation to any dimension of spacetime and to any spacetime theory of gravity.
- the users who have little or no knowledge of Mathematica syntax or xAct syntax can derive the same utility from the package pretty much the same way, an expert in those packages would. T
- The package is relatively fast. At first order, the entire equations of General relativity could be derived in less than a second. As the order in perturbation theory increases, the timing grows as a power law .

Finally, we plan to extend this approach beyond derivation of equations of general relativity or the conservation of energy-momentum tensor equation to entire Einstein-Boltmann systems and Einstein-Jacobi map system needed for understanding weak lensing and large scale galaxy clustering.

ACKNOWLEDGMENTS

It is a pleasure to thank José-Maria Martìn-Garcìa and Guillaume Faye for long and fruitful discussions about abstract tensor calculus in *xAct* [4].

---

[1] E. Komatsu *et al.* (WMAP), Astrophys. J. Suppl. **180**, 330 (2009), arXiv:0803.0547 [astro-ph].
[2] D. P. Peter Musgrave and K. Lake, “”grtensorii”,” (1996).
[3] D. Brizuela, J. M. Martin-Garcia, and G. A. Mena Marugan, Gen.Rel.Grav. **41**, 2415 (2009), arXiv:0807.0824 [gr-qc].
[4] J. Martin-Garcìa, “”xact and xpert”,” (2004).
[5] D. Brizuela, J. M. Martin-Garcia, and G. A. Mena Marugan, Phys. Rev. **D74**, 044039 (2006), arXiv:gr-qc/0607025.
[6] D. Brizuela, J. M. Martin-Garcia, and M. Tiglio, Phys.Rev. **D80**, 024021 (2009), arXiv:0903.1134 [gr-qc].
[7] D. Brizuela, J. M. Martin-Garcia, U. Sperhake, and K. D. Kokkotas, Phys.Rev. **D82**, 104039 (2010), arXiv:1009.5605 [gr-qc].
[8] R. L. Arnowitt, S. Deser, and C. W. Misner, (1962), arXiv:gr-qc/0405109 [gr-qc].
[9] M. Bruni, S. Matarrese, S. Mollerach, and S. Sonego, Class. Quant. Grav. **14**, 2585 (1997).
[10] K. Nakamura, Prog. Theor. Phys. **117**, 17 (2007).
[11] R. M. Wald, *General relativity* (University of Chicago Press, Chicago, 1984).
[12] E. Gourgoulhon, (2007), lecture notes, arXiv:gr-qc/0703035 [GR-QC].
[13] G. Ellis, Gen.Rel.Grav. **41**, 581 (2009).
[14] G. F. R. Ellis and H. van Elst, NATO Adv. Study Inst. Ser. C. Math. Phys. Sci. **541**, 1 (1999), arXiv:gr-qc/9812046.
[15] K. A. Malik and D. Wands, Phys.Rept. **475**, 1 (2009), arXiv:0809.4944 [astro-ph].
[16] J. M. Bardeen, Phys.Rev. **D22**, 1882 (1980).
[17] M. Bruni, S. Matarrese, S. Mollerach, and S. Sonego, Class.Quant.Grav. **14**, 2585 (1997), arXiv:gr-qc/9609040 [gr-qc].
[18] K. A. Malik, (2001), arXiv:astro-ph/0101563 [astro-ph].
[19] K. Nakamura, Prog.Theor.Phys. **117**, 17 (2007), arXiv:gr-qc/0605108 [gr-qc].
[20] T. S. Pereira, C. Pitrou, and J.-P. Uzan, JCAP **0709**, 006 (2007), arXiv:0707.0736 [astro-ph].
[21] L. Abbott and R. K. Schaefer, Astrophys.J. **308**, 546 (1986).
[22] C. Pitrou, T. S. Pereira, and J.-P. Uzan, JCAP **0804**, 004 (2008), arXiv:0801.3596 [astro-ph].
[23] T. S. Pereira, S. Carneiro, and G. A. M. Marugan, JCAP **1205**, 040 (2012), arXiv:1203.2072 [astro-ph.CO].
[24] G. F. R. Ellis and M. A. H. MacCallum, Communications in Mathematical Physics **12**, 108 (1969).
[25] A. Pontzen and A. Challinor, Mon.Not.Roy.Astron.Soc. (2007), arXiv:0706.2075 [astro-ph].

Appendix A: Classification of homogeneous cosmologies

In this appendix, we review the general properties of Bianchi Scape-times. These have homogeneous hypersurfaces, and thus there exist by construction three linearly independent Killing vector fields (KVF) $\boldsymbol{\xi}_i$, with $i \in \{1, 2, 3\}$, satisfying [14]:

$$\mathcal{L}_{\boldsymbol{\xi}_i} \bar{g}_{\mu\nu} = 0 \quad \Leftrightarrow \quad \bar{\nabla}_{(\mu}\, \xi_{i\ \nu)} = 0\,, \qquad \bar{u}_\mu \xi_i^\mu = 0\,. \tag{A1}$$

From these relations and upon considering the vanishing of the vorticity of $\bar{\mathbf{n}}$ (equation (25)), we obtain:

$$\mathcal{L}_{\bar{\boldsymbol{n}}} \boldsymbol{\xi}_i = [\bar{\boldsymbol{n}}, \boldsymbol{\xi}_i] = 0\,, \tag{A2}$$

and hence:

$$\mathcal{L}_{\boldsymbol{\xi}_i} \bar{h}_{\mu\nu} = 0\,. \tag{A3}$$

The nature of the spatially homogeneous model is determined by the structure coefficients $C^k{}_{ij}$ (with i, j and k running in $\{1, 2, 3\}$), defined by the commutators of the KVF [11, 24]:

$$[\boldsymbol{\xi}_i, \boldsymbol{\xi}_j] \equiv -C^k{}_{ij} \boldsymbol{\xi}_k\,, \qquad \text{with} \qquad C^k{}_{ij} = -C^k{}_{ji}\,. \tag{A4}$$

Substituting this into the Jacobi identity,

$$\big[\boldsymbol{\xi}_i,[\boldsymbol{\xi}_j,\boldsymbol{\xi}_k]\big]+\big[\boldsymbol{\xi}_j,[\boldsymbol{\xi}_k,\boldsymbol{\xi}_i]\big]+\big[\boldsymbol{\xi}_k,[\boldsymbol{\xi}_i,\boldsymbol{\xi}_j]\big]=0\,, \tag{A5}$$

we find the requirement

$$C^m{}_{[ij}C^l{}_{k]m}=0 \quad\Rightarrow\quad C^m{}_{ij}C^l{}_{lm}=0\,, \tag{A6}$$

where we have used the fact that the structure coefficients are constant on the spatial slices.

We now construct on a given hypersurface a vector basis $\{\mathbf{e}_i\}$ and its dual basis $\{\mathbf{e}^i\}$, invariant under the action of the KVF:

$$\mathcal{L}_{\boldsymbol{\xi}_i}\mathbf{e}_j=[\boldsymbol{\xi}_i,\mathbf{e}_j]=0\,,\qquad \mathcal{L}_{\boldsymbol{\xi}_i}\mathbf{e}^j=0\,, \tag{A7}$$

From these properties along with relation (A4), we deduce:

$$[\mathbf{e}_i,\mathbf{e}_j]=C^k{}_{ij}\mathbf{e}_k\,,\qquad 2e_i{}^\mu e_j{}^\nu\nabla_{[\mu}e^k{}_{\nu]}=-C^k{}_{ij}\,, \tag{A8}$$

where ϵ_{ijm} denotes the Levi–Civita symbol. The coefficients of structure can be further developed in terms of a symmetric tensor n^{ij} and a 'vector' a^i as

$$C^k{}_{ij}=\epsilon_{ijm}N^{mk}+2A_{[i}\delta^k{}_{j]}\,. \tag{A9}$$

The classification of Bianchi space-times is exposed in details notably in Ref. [24] and also summarized briefly in Ref. [25]. The bases $\{\mathbf{e}_i\}$ and $\{\mathbf{e}^i\}$ are then extended to the whole space-time by Lie dragging them with $\bar{\mathbf{n}}$:

$$\mathcal{L}_{\bar{\mathbf{n}}}\mathbf{e}_i=[\bar{\mathbf{n}},\mathbf{e}_i]=0\,,\qquad \mathcal{L}_{\bar{\mathbf{n}}}\mathbf{e}^i=0\,. \tag{A10}$$

With the above prescription, we are thus able to construct a four-dimensional basis $\{\mathbf{e}_a\}\equiv\{\bar{\mathbf{n}},\mathbf{e}_i\}$ along with its dual $\{\mathbf{e}^a\}\equiv\{\underline{\bar{\mathbf{n}}},\mathbf{e}^i\}$ (where $\underline{\bar{\mathbf{n}}}$ is the dual form of $\bar{\mathbf{n}}$ and $a\in\{0,1,2,3\}$), that are invariant under the action of the KVF. The commutation relations simply follow from expressions (A8) and (A10): the structure coefficients $C^c{}_{ab}$ vanish when any of the indices is zero and takes the values $C^k{}_{ij}$ otherwise. This method to build a four-dimensional basis out of a three-dimensional one, defined on a given spatial hypersurface, is the simplest one[3]

The components of the induced metric of the hypersurfaces can then be written as

$$\bar{h}_{\mu\nu}=\bar{h}_{ij}e^i{}_\mu e^j{}_\nu\,. \tag{A11}$$

From expressions (A3) and (A7), we have: $\mathbf{e}_k(h_{ij})=0$; and hence the components h_{ij} are only time-dependent, $h_{ij}=h_{ij}(\eta)$. The connection coefficients associated with the background metric are given in this basis by

$$\bar{\Gamma}_{abc}=\frac{1}{2}\big(-\mathbf{e}_a\left(\bar{g}_{bc}\right)+\mathbf{e}_b\left(\bar{g}_{ca}\right)+\mathbf{e}_c\left(\bar{g}_{ab}\right)+C_{abc}-C_{bca}+C_{cab}\big)\,. \tag{A12}$$

We note that the spatial connection coefficients $\bar{\Gamma}_{ijk}$ are only expressed in terms of the constants of structure since, as mentioned above, the components $\bar{h}_{ij}$ only depend on η. Thus we have:

$$\bar{\Gamma}_{ijk}=\frac{1}{2}\big(C_{ijk}-C_{jki}+C_{kij}\big)\,,\qquad C_{ijk}=\bar{\Gamma}_{ijk}-\bar{\Gamma}_{ikj}\,. \tag{A13}$$

Note that indices in the basis of the $\boldsymbol{e}_i$ and $\boldsymbol{e}^i$ are lowered with $\bar{h}_{ij}$ and raised with its inverse $\bar{h}^{ij}$, so that for instance $C_{kij}\equiv\bar{h}_{km}C^m{}_{ij}$. Furthermore, any tensor field on the background space-time must have the same symmetries. For instance a homogenous tensor is necessarily of the form

$$\bar{V}_{\nu_1\dots\nu_p}=\bar{V}_{i_1\dots i_p}(\eta)e^{i_1}{}_{\nu_1}\dots e^{i_p}{}_{\nu_p}\,, \tag{A14}$$

from which we deduce that

$$\bar{D}_k\bar{V}_{i_1\dots i_p}\equiv e_k{}^\alpha e_{i_1}{}^{\nu_1}\dots e_{i_p}{}^{\nu_p}\bar{D}_\alpha\bar{V}_{\nu_1\dots\nu_p}=-\sum_{j=1}^{p}\bar{\Gamma}^q_{k\,i_j}\bar{V}_{i_1\dots i_{j-1}\,q\,i_{j+1}\dots i_p}\,. \tag{A15}$$

---

[3] Note that in this framework, only $\bar{\mathbf{n}}$ is a unit vector. An alternative approach consists in building a basis of vectors that all are unitary, by renormalizing the $\mathbf{e}_i$. However, by doing so, the obtained spatial vectors do not commute with $\bar{\mathbf{n}}$ anymore, and their associated structure coefficients become time-dependent. We shall not consider such possibility in the present paper, but details can be found in, e.g., [14, 24, 25].

This relation is essentially used to compute the induced covariant derivative of the extrinsic curvature, since it is a background tensor. Note also that the constants of structure are in fact the components in this specific basis of a tensor, and in general the tensor associated can be recovered from

$$C^{\alpha}{}_{\mu\nu} \equiv C^{k}{}_{ij} e_k{}^{\alpha} e^{i}{}_{\mu} e^{j}{}_{\nu} \,. \tag{A16}$$

Similarly, the tensors associated with n_{ij} and a_i are found from

$$n_{\mu\nu} \equiv n_{ij} e^{i}{}_{\mu} e^{j}{}_{\nu}, \quad a_\mu \equiv a_i e^{i}{}_{\mu} \,. \tag{A17}$$

So the relation A15 is actually used to compute the induced covariant derivative of these tensors since they also live on the background space-time.

The Riemann tensor of the induced metric can be expressed only in terms of the constants of structure. In the basis of the $\boldsymbol{e}_i$ and $\boldsymbol{e}^i$ its components are given by

$${}^3\bar{R}_{ij}{}^{kl} = -\frac{1}{2} C^{p}{}_{ij} C_{p}{}^{kl} + \frac{1}{2} C_{p}{}^{l}{}_{i} C^{pk}{}_{j} + C_{p}{}^{l}{}_{j} C_{i}{}^{kp} + C_{p}{}^{l}{}_{j} C^{k}{}_{i}{}^{p} + C_{ijp} C^{pkl} + \frac{1}{2} C_{i}{}^{l}{}_{p} C_{j}{}^{kp} + \frac{1}{2} C^{l}{}_{ip} C^{k}{}_{j}{}^{p} + C^{k}{}_{jp} C_{i}{}^{lp} \,. \tag{A18}$$

where we remind again that the indices on $C^{k}{}_{ij}$ are lowered and raised with $\bar{h}_{ij}$ and its inverse $\bar{h}^{ij}$, and where a double antisymmetrization $[ij]$ and $[kl]$ is implied on the indices in the right hand side. The Ricci tensor can then be deduced and we obtain

$$\begin{aligned} {}^3\bar{R}_{ij} &= -\frac{1}{2} C_{kil} C^{k}{}_{j}{}^{l} - \frac{1}{2} C_{kil} C^{l}{}_{j}{}^{k} + \frac{1}{4} C_{i}{}^{kl} C_{jkl} + C_{(ij)}{}^{p} C^{k}{}_{pk} \\ {}^3\bar{R} &= -\frac{1}{4} C_{ijk} C^{ijk} - \frac{1}{2} C_{ijk} C^{jik} + C^{kj}{}_{k} C^{p}{}_{pj} \end{aligned} \tag{A19}$$

Note that thanks to the Jacobi identities (A6), the Riemann and Ricci tensors of the induced metric can take several equivalent forms. Finally, since the hypersurfaces are homogeneous, any induced derivative on ${}^3\bar{R}_{ijkl}$ can be computed using Eq. (A15). All the rules of this appendix, that is the equations (A15),(A13),(A18) and (A19) are automatically implemented when calling the function `BackgroundSlicing` in case the space type specified is of Bianchi type. An option boolean controls whether or not the constantsof structure should be opened using the parameterization (A9) in the final expressions.