

## Lecture 10: The Perturbation module

# Perturbations

Goal:

- to **integrate** the differential system of cosmological perturbations

# Perturbations

## Goal:

- to **integrate** the differential system of cosmological perturbations
- to **store** several functions  $S_i(k, \tau)$  that might be needed in other modules

# Perturbations

Goal:

- to **integrate** the differential system of cosmological perturbations
- to **store** several functions  $S_i(k, \tau)$  that might be needed in other modules
- immediately useful for output in Fourier space (e.g.:  $\delta_{\text{matter}}(k, \tau)$  for  $P(k, z)$ ).

# Perturbations

## Goal:

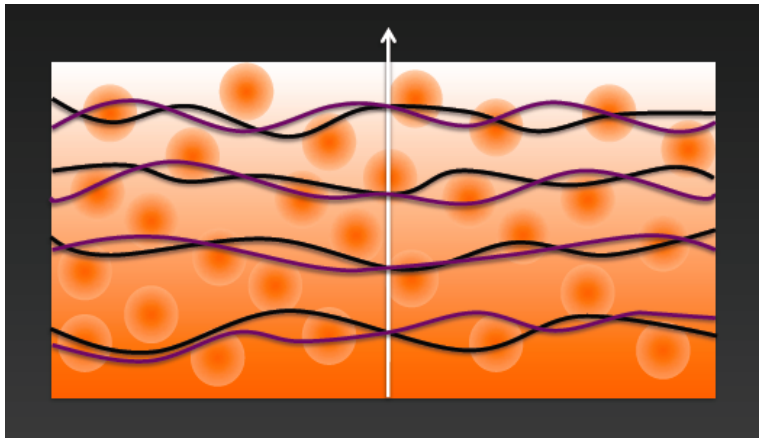
- to **integrate** the differential system of cosmological perturbations
- to **store** several functions  $S_i(k, \tau)$  that might be needed in other modules
- immediately useful for output in Fourier space (e.g.:  $\delta_{\text{matter}}(k, \tau)$  for  $P(k, z)$ ).
- first step for the  $C_l$ 's, which depend on particular sources  $S_i(k, \tau)$  integrated over time.

# Perturbations

## Goal:

- to **integrate** the differential system of cosmological perturbations
- to **store** several functions  $S_i(k, \tau)$  that might be needed in other modules
- immediately useful for output in Fourier space (e.g.:  $\delta_{\text{matter}}(k, \tau)$  for  $P(k, z)$ ).
- first step for the  $C_l$ 's, which depend on particular sources  $S_i(k, \tau)$  integrated over time.

In this module, everything is assumed to be linear.



- One gauge = one way to slice the space-time in equal-time hypersurfaces.
- Any time slicing such that quantities vary perturbatively on each slice is valid.
- Fixing the gauge = imposing a restriction (e.g. on number of non-zero metric perturbations) such that time slicing is fixed.
- Any choice of gauge leads to same observable quantities.

$\delta g_{\mu\nu}$ : 10 d.o.f., 4 scalars, 4 vectors, 2 tensors.

For scalar sector:

- imposing zero non-diagonal terms fixes the gauge.

$$ds^2 = -(1 + 2\psi)dt^2 + (1 - 2\phi)a^2 d\vec{l}^2.$$

Newtonian gauge.

- imposing zero terms in  $\delta g_{00}$  and  $\delta g_{0i}$  leaves also 2 d.o.f ( $h', \eta$ ), but does not fix the gauge.

But extra condition  $\theta_i(k, \tau_{\text{ini}}) = 0$  does.

Synchronous gauge comoving with species  $i$ .

For a pressureless component,  $\theta'_i = -\frac{a'}{a}\theta_i$ . Hence usually use the synchronous gauge comoving with CDM: saves one equation (but one more dynamical Einstein equation).

CAMB and CMBFAST use the synchronous gauge comoving with CDM. In CLASS user can choose:

gauge = synchronous    or    gauge = newtonian



# Equations of motion

**Fluids** get simple equations of motion because interactions impose unique value of isotropic pressure in each point. Described by  $\delta = \delta\rho/\rho$ ,  $\theta$ ,  $\delta p$ . For perfect fluids  $\delta p = w\delta\rho$ . By extension we allow for  $\delta p = c_s^2\delta\rho$  with a constant  $c_s^2 \neq w$ . For decoupled non-relativistic species, pressure perturbations can be neglected, so effectively equivalent to fluid with  $c_s^2 = 0$ .

- $\delta' = -(1+w)\theta - 3\frac{a'}{a}(c_s^2 - w)\delta + \text{metric\_continuity}$  (continuity)
- $\theta' = -\frac{a'}{a}(1-3w)\theta - \frac{w'}{1+w}\theta + \frac{k^2 c_s^2}{1+w}\delta + \text{metric\_euler}$  (Euler)

# Equations of motion

For **streaming particles**: phase space,  $f = \bar{f}(\tau, p) + \delta f(\tau, p, \vec{x}, \hat{n})$ .

Get  $\delta f$  evolution from Boltzmann equation.

Initially in thermal equilibrium:  $\delta f$  related to  $\Theta(\tau, \vec{x})$ , simple Boltzmann equation equivalent to continuity+Euler.

Next:

$$\frac{d \ln(ap)}{d\tau} = -\phi' - \frac{\sqrt{p^2 + m^2}}{p} \hat{n} \cdot \vec{\nabla} \psi.$$

- **ultra-relativistic**: temperature still works but  $\Theta(\tau, \vec{x}, \hat{n})$  (hierarchy in multipole space,  $F_l$ ).
- **non-relativistic**: use  $f = \bar{f}(1 + \Psi(\tau, p, \vec{x}, \hat{n}))$ , Boltzmann in momentum space.

# Equations of motion

Einstein equations: 4 scalar equations, but redundant with equations of motion of species (Bianchi identity).

- **Newtonian gauge:** can be used as constraint equation.
- **synchronous gauge comoving with CDM:** cannot eliminate differential operator. One dynamical equation, one constraint equation.

In total, same number of equation. However, for high precision in Newtonian gauge, better to replace one constraint equation by one dynamical equation.

Whole ODE: fluid equations, Boltzmann hierarchies, one Einstein equation.

# Initial conditions

Different choices (adiabatic, baryon isocurvature, CDM isocurvature, etc..) will be reviewed tomorrow. In each of these cases, all perturbations relate to a **single quantity** at initial time:

$$\forall i, \quad A_i(\tau_{\text{ini}}, \vec{k}) = a_i \mathcal{R}(\tau_{\text{ini}}, \vec{k}).$$

At later time, linearity + isotropy impose

$$\forall i, \quad A_i(\tau, \vec{k}) = A_i(\tau, k) \mathcal{R}(\tau_{\text{ini}}, \vec{k}).$$

Here  $A_i(\tau, k)$  is the transfer function normalized to  $\mathcal{R}$ . Given by the solution of ODE normalized to  $\mathcal{R}(\tau_{\text{ini}}, \vec{k}) = 1$ . Power spectra:

$$\left\langle \left| A_i(\tau, \vec{k}) \right|^2 \right\rangle = (A_i(\tau, k))^2 \left\langle \left| \mathcal{R}(\tau_{\text{ini}}, \vec{k}) \right|^2 \right\rangle.$$

**Separate tasks** for the **perturbation module**, **primordial module** and **spectra module**.

# Line-of-sight integral

Boltzmann hierarchy for photons or massless neutrinos:  $\left\{ \begin{array}{l} F'_0 = \dots \\ F'_1 = \dots \\ \dots \\ F'_l = \dots \end{array} \right.$

(Ma & Bertschinger:  $F_l = 4\Theta_l$ )

# Line-of-sight integral

Boltzmann hierarchy for photons or massless neutrinos: 
$$\left\{ \begin{array}{l} F'_0 = \dots \\ F'_1 = \dots \\ \dots \\ F'_l = \dots \end{array} \right.$$

(Ma & Bertschinger:  $F_l = 4\Theta_l$ )

One more hierarchy  $G_l$  to account for polarisation (i.e. for Stokes parameters, given symmetry of Thomson scattering term)

# Line-of-sight integral

Boltzmann hierarchy for photons or massless neutrinos: 
$$\begin{cases} F'_0 & = & \dots \\ F'_1 & = & \dots \\ & \dots & \\ F'_l & = & \dots \end{cases}$$

(Ma & Bertschinger:  $F_l = 4\Theta_l$ )

One more hierarchy  $G_l$  to account for polarisation (i.e. for Stokes parameters, given symmetry of Thomson scattering term)

Power spectrum in multipole space:  $C_l^{TT} = 4\pi \int \frac{dk}{k} (\Theta_l(\tau_0, k))^2 \mathcal{P}(k)$ . A priori, we need to integrate the Boltzmann hierarchy up to high  $l$ 's.

# Line-of-sight integral

Boltzmann hierarchy for photons or massless neutrinos: 
$$\left\{ \begin{array}{l} F'_0 = \dots \\ F'_1 = \dots \\ \dots \\ F'_l = \dots \end{array} \right.$$

(Ma & Bertschinger:  $F_l = 4\Theta_l$ )

One more hierarchy  $G_l$  to account for polarisation (i.e. for Stokes parameters, given symmetry of Thomson scattering term)

Power spectrum in multipole space:  $C_l^{TT} = 4\pi \int \frac{dk}{k} (\Theta_l(\tau_0, k))^2 \mathcal{P}(k)$ . A priori, we need to integrate the Boltzmann hierarchy up to high  $l$ 's.

But clever integration by part of Boltzmann equation gives

$$\Theta_l(\tau_0, k) = \int_{\tau_{\text{ini}}}^{\tau_0} d\tau S_T(\tau, k) j_l(k(\tau_0 - \tau))$$

$$S_T(\tau, k) \equiv \underbrace{g(\Theta_0 + \psi)}_{\text{SW}} + \underbrace{(g k^{-2} \theta_b)'}_{\text{Doppler}} + \underbrace{e^{-\kappa}(\phi' + \psi')}_{\text{ISW}} + \text{polarisation} .$$

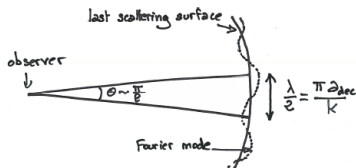


# Line-of-sight integral

$$\Theta_l(\tau_0, k) = \int_{\tau_{\text{ini}}}^{\tau_0} d\tau S_T(\tau, k) j_l(k(\tau_0 - \tau))$$

$$S_T(\tau, k) \equiv \underbrace{g(\Theta_0 + \psi)}_{\text{SW}} + \underbrace{(g k^{-2} \theta_b)'}_{\text{Doppler}} + \underbrace{e^{-\kappa}(\phi' + \psi')}_{\text{ISW}} + \text{polarisation} .$$

Role of Bessel: projection from Fourier to harmonic space



$$\theta d_a(z_{\text{rec}}) = \frac{\lambda}{2} \text{ gives precisely } l = k(\tau_0 - \tau_{\text{rec}})$$

Instead of calculating all  $\Theta_l(\tau, k)$  and store them, we only need to calculate  $\Theta_{l \leq 10}(\tau, k)$  and store  $S_T(k, \tau)$ .

However this source is a non-trivial function ( $\psi'$  and polarisation terms)...

# Line-of-sight integral

Full source function in CAMB:

```
!Maple fortran output - see scal_eqs.map
  ISW = (4.D0/3.D0*k*EV%Kf(1)*sigma+(-2.D0/3.D0*sigma
    -2.D0/3.D0*etak/adotoa)*k &
    -diff_rhopi/k**2-1.D0/adotoa*dgrho/3.D0+(3.D0*
      gpres+5.D0*grho)*sigma/k/3.D0 &
    -2.D0/k*adotoa/EV%Kf(1)*etak)*expmmu(j)
!The rest, note y(9)->octg, yprime(9)->octgprime (octopoles)
sources(1)= ISW + ((-9.D0/160.D0*pig-27.D0/80.D0*ypol
  (2))/k**2*opac(j)+(11.D0/10.D0*sigma- &
  3.D0/8.D0*EV%Kf(2)*ypol(3)+vb-9.D0/80.D0*EV%Kf(2)*octg
  +3.D0/40.D0*qg)/k-(- &
  180.D0*ypolprime(2)-30.D0*pigdot)/k**2/160.D0)*dvis(j)
  +(-(9.D0*pigdot+ &
  54.D0*ypolprime(2))/k**2*opac(j)/160.D0+pig/16.D0+clxg
  /4.D0+3.D0/8.D0*ypol(2))+(- &
  21.D0/5.D0*adotoa*sigma-3.D0/8.D0*EV%Kf(2)*ypolprime(3)+
  vbdot+3.D0/40.D0*qgdot- &
  9.D0/80.D0*EV%Kf(2)*octgprime)/k+(-9.D0/160.D0*dopac(j)*
  pig-21.D0/10.D0*dgpi-27.D0/ &
  80.D0*dopac(j)*ypol(2))/k**2)*vis(j)+(3.D0/16.D0*ddvis(j)
  )*pig+9.D0/ &
  8.D0*ddvis(j)*ypol(2))/k**2+21.D0/10.D0/k/EV%Kf(1)*vis(j)
  )*etak
```



# Line-of-sight integral

$S_T(k, \tau)$  comes from integration by part of

$$\Theta_l(\tau_0, k) = \int_{\tau_{\text{ini}}}^{\tau_0} d\tau \left\{ S_T^0(\tau, k) j_l(k(\tau_0 - \tau)) \right. \\ \left. + S_T^1(\tau, k) \frac{dj_l}{dx}(k(\tau_0 - \tau)) \right. \\ \left. + S_T^2(\tau, k) \frac{1}{2} \left[ 3 \frac{d^2 j_l}{dx^2}(k(\tau_0 - \tau)) + j_l(k(\tau_0 - \tau)) \right] \right\}$$

CLASS v2.0 stores separately  $S_T^1(\tau, k)$ ,  $S_T^2(\tau, k)$ ,  $S_T^2(\tau, k)$ , and the transfer module will convolve them individually with respective bessel functions.

$$S_T^0 = g \left( \frac{\delta g}{4} + \psi \right) + e^{-\kappa} (\phi' + \psi') \quad S_T^1 = g \frac{\theta_b}{k} \quad S_T^2 = \frac{g}{8} (G_0 + G_2 + F_2)$$

or

$$S_T^0 = g \left( \frac{\delta g}{4} + \phi \right) + e^{-\kappa} 2\phi' + g'\theta_b + g\theta_b' \quad S_T^1 = e^{-\kappa} k(\psi - \phi) \quad S_T^2 = \frac{g}{8} (G_0 + G_2 + F_2)$$

# Tensor perturbations

- **tensor perturbations** present in non-diagonal spatial part of the metric  $\delta g_{\mu\nu}$  (2 d.o.f. of GW) and of  $\delta T_{\mu\nu}$  of streaming component (decoupled neutrinos and photons).
- no gauge ambiguity for tensors.
- **Einstein equations** are of course dynamical for GW. For each of the two polarisation states,

$$h'' + 2\frac{a'}{a}h' + k^2h = f[F_l^\gamma, F_l^\nu] .$$

- need to integrate also **Boltzmann hierarchies for photons and neutrinos**, sourced by GW fields.
- there is also a **line-of-sight approach** for tensors,

$$\Theta_l(\tau_0, k) = \int_{\tau_{\text{ini}}}^{\tau_0} d\tau S_T(\tau, k) \sqrt{\frac{3(l+2)!}{8(l-2)!}} \frac{j_l(k(\tau_0 - \tau))}{(k(\tau_0 - \tau))^2}$$

$$S_T(\tau, k) = -g \left( \frac{F_{\gamma 0}^{(2)}}{10} + \frac{F_{\gamma 2}^{(2)}}{7} + \frac{3F_4^{(2)}}{70} - \frac{3G_{\gamma 0}^{(2)}}{5} + \frac{6G_{\gamma 2}^{(2)}}{7} - \frac{3G_{\gamma 4}^{(2)}}{70} \right) e^{-\kappa h'}$$

# Polarisation

- Created by **Thomson scattering** when photons are NOT in thermal equilibrium anymore (recombination, reionisation), such that their "environnement" has a **quadrupolar component**.
- E-mode and B-mode can be derived from a **single source**  $S_P(\tau, k)$ , convolved with different *radial functions* in transfer module.
- absence of any integrated-Sachs-Wolfe-like term:

$$S_P(\tau, k) = \sqrt{6} g(\tau) P(\tau, k)$$

For scalars:

$$P = \frac{1}{8} \left[ F_{\gamma^2}^{(0)} + G_{\gamma^0}^{(0)} + G_{\gamma^2}^{(0)} \right]$$

For tensors:

$$P = -\frac{1}{\sqrt{6}} \left[ \frac{F_{\gamma^0}^{(2)}}{10} + \frac{F_{\gamma^2}^{(2)}}{7} + \frac{3F_4^{(2)}}{70} - \frac{3G_{\gamma^0}^{(2)}}{5} + \frac{6G_{\gamma^2}^{(2)}}{7} - \frac{3G_{\gamma^4}^{(2)}}{70} \right]$$

Can be used also in non-flat space: T. Tram & JL, JCAP 2013 [arXiv:1305.3261]

# Which sources?

Computing/storing each of these sources is decided automatically by the code depending on what the user asked:

- which output? CMB temperature, CMB polarisation, CMB lensing, matter power spectrum, all densities  $\{\delta_i\}$ , all velocities  $\{\theta_i\}$ ,  $C_l$ 's of number count in redshift bin,  $C_l$ 's of cosmic shear in redshift bin, corresponding respectively to output = tCl, pCl, lCl, mPk, dTk, vTk, nCl, sCl

# Which sources?

Computing/storing each of these sources is decided automatically by the code depending on what the user asked:

- which output? CMB temperature, CMB polarisation, CMB lensing, matter power spectrum, all densities  $\{\delta_i\}$ , all velocities  $\{\theta_i\}$ ,  $C_l$ 's of number count in redshift bin,  $C_l$ 's of cosmic shear in redshift bin, corresponding respectively to `output = tCl, pCl, lCl, mPk, dTk, vTk, nCl, sCl`
- which modes? scalars, vectors, tensors, associated to indices `index_md_scalars, index_md_vectors, index_md_tensors`.

# Which sources?

Computing/storing each of these sources is decided automatically by the code depending on what the user asked:

- which output? CMB temperature, CMB polarisation, CMB lensing, matter power spectrum, all densities  $\{\delta_i\}$ , all velocities  $\{\theta_i\}$ ,  $C_l$ 's of number count in redshift bin,  $C_l$ 's of cosmic shear in redshift bin, corresponding respectively to `output = tCl, pCl, lCl, mPk, dTk, vTk, nCl, sCl`
- which modes? scalars, vectors, tensors, associated to indices `index_md_scalars, index_md_vectors, index_md_tensors`.
- Different number of sources for each mode, `tp_size[index_mode]`



# Which sources?

Computing/storing each of these sources is decided automatically by the code depending on what the user asked:

- which output? CMB temperature, CMB polarisation, CMB lensing, matter power spectrum, all densities  $\{\delta_i\}$ , all velocities  $\{\theta_i\}$ ,  $C_l$ 's of number count in redshift bin,  $C_l$ 's of cosmic shear in redshift bin, corresponding respectively to `output = tCl, pCl, lCl, mPk, dTk, vTk, nCl, sCl`
- which modes? scalars, vectors, tensors, associated to indices `index_md_scalars, index_md_vectors, index_md_tensors`.
- Different number of sources for each mode, `tp_size[index_mode]`
- The code must consider  $S(k, \tau)$  for each type, each mode, but also each initial condition (adiabatic, CDM isocurvature, neutrino isocurvature, etc., again associated to indices).

# Which sources?

Computing/storing each of these sources is decided automatically by the code depending on what the user asked:

- which output? CMB temperature, CMB polarisation, CMB lensing, matter power spectrum, all densities  $\{\delta_i\}$ , all velocities  $\{\theta_i\}$ ,  $C_l$ 's of number count in redshift bin,  $C_l$ 's of cosmic shear in redshift bin, corresponding respectively to `output = tCl, pCl, lCl, mPk, dTk, vTk, nCl, sCl`
- which modes? scalars, vectors, tensors, associated to indices `index_md_scalars, index_md_vectors, index_md_tensors`.
- Different number of sources for each mode, `tp_size[index_mode]`
- The code must consider  $S(k, \tau)$  for each type, each mode, but also each initial condition (adiabatic, CDM isocurvature, neutrino isocurvature, etc., again associated to indices).
- could define

```
ppt->sources[index_md][index_ic][index_type][index_tau][index_k]  
but better to use
```

```
ppt->sources[index_md][index_ic * ppt->tp_size[index_md] +  
index_type][index_tau * ppt->k_size + index_k]
```

# Which sources?

- sources for CMB temperature (decomposed in 1 to 3 terms)
- sources for CMB polarisation (only 1 term)
- sum of metric perturbations  $\phi + \psi$  (useful for lensing)
- density perturbations of total non-relativistic matter  $\delta_m$  (for  $P(k)$ , NC)
- $\phi$ ,  $\psi$ ,  $\phi'$ , total velocity of non-relativistic matter  $\theta_m$  (for NC)
- density perturbations of all components with non-zero density  $\{\delta_i\}$
- velocity perturbations of all components with non-zero density  $\{\theta_i\}$

These different type of sources are called **perturbation types** and associated to indices

- `index_tp_t0`, `index_tp_t1`, `index_tp_t2`,
- `index_tp_p`,
- `index_tp_phi_plus_psi`, `index_tp_phi`, `index_tp_psi`, `index_tp_phi_prime`
- `index_tp_delta_m`, `index_tp_theta_m`
- `index_tp_delta_g`, `index_tp_delta_cdm`, etc.
- `index_tp_theta_g`, `index_tp_theta_cdm`, etc.

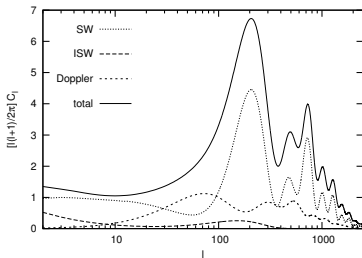
of size `tp_size[index_mode]`.

# Very few physical input parameters in this module...

- Possibility to take into account only a few contribution to the sources:

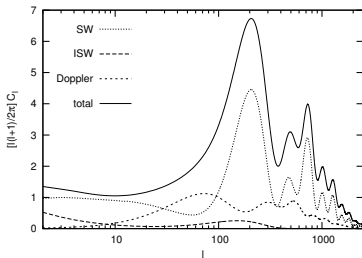
# Very few physical input parameters in this module...

- Possibility to take into account only a few contribution to the sources:
  - For CMB temperature  $tCl$  (to understand physically different effects):  
temperature contributions = tsw, eisw, lisw, dop, pol  
early/late isw redshift = 50



# Very few physical input parameters in this module...

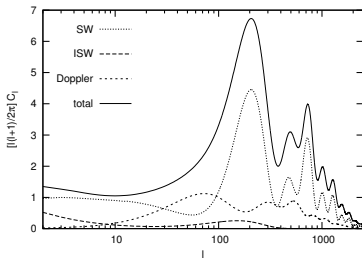
- Possibility to take into account only a few contribution to the sources:
  - For CMB temperature  $tCl$  (to understand physically different effects):  
temperature contributions = tsw, eisw, lisw, dop, pol  
early/late isw redshift = 50



- For number count  $nCl$  (to understand physically different effects, and also to speed up, keeping only leading contributions, see arxiv:1307.1459 or Bonvin & Durrer arxiv:1105.5280):  
number count contributions = density, rsd, lensing, gr

# Very few physical input parameters in this module...

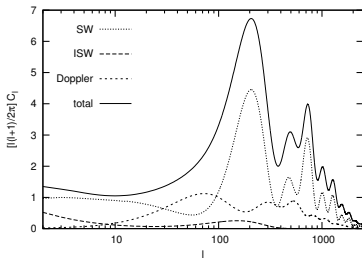
- Possibility to take into account only a few contribution to the sources:
  - For CMB temperature  $tCl$  (to understand physically different effects):  
temperature contributions = tsw, eisw, lisw, dop, pol  
early/late isw redshift = 50



- For number count  $nCl$  (to understand physically different effects, and also to speed up, keeping only leading contributions, see arxiv:1307.1459 or Bonvin & Durrer arxiv:1105.5280):  
number count contributions = density, rsd, lensing, gr
- perturbed recombination = [no/yes] for advanced modelling of baryon fluctuations during recombination

# Very few physical input parameters in this module...

- Possibility to take into account only a few contribution to the sources:
  - For CMB temperature  $tCl$  (to understand physically different effects):  
temperature contributions = tsw, eisw, lisw, dop, pol  
early/late isw redshift = 50



- For number count  $nCl$  (to understand physically different effects, and also to speed up, keeping only leading contributions, see arxiv:1307.1459 or Bonvin & Durrer arxiv:1105.5280):  
number count contributions = density, rsd, lensing, gr
- perturbed recombination = [no/yes] for advanced modelling of baryon fluctuations during recombination
- gauge = [synchronous/newtonian]



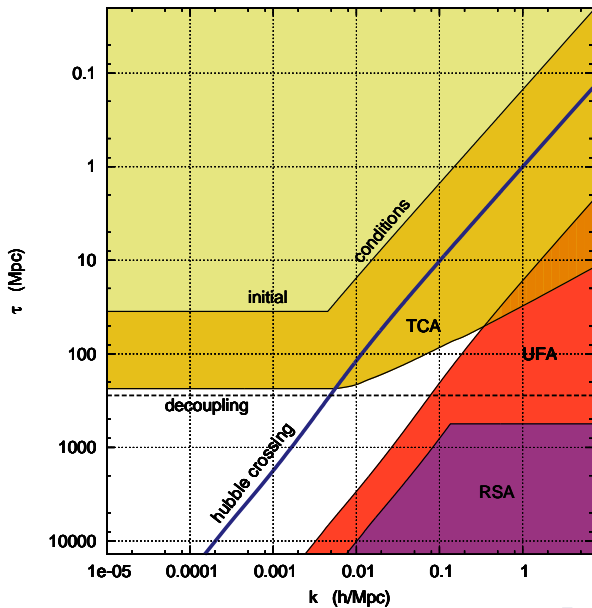
# Main tasks in `perturb_init()`

- define all indices with `perturb_indices()`
- find an optimal time-sampling of the sources, based on the variation rate of background and thermodynamical quantities, and on  $k$ .
- loop over all modes, initial conditions, wavenumbers
- for each of them, call `perturb_solve()` to compute  $S(k, \tau)$  of each type

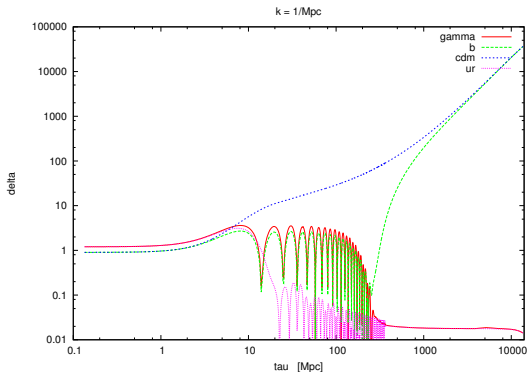
# Main tasks in `perturb_solved()`

- to find when approximation schemes (tight coupling, ultra-relativistic fluid approximation, radiation streaming approximation) must be switched off or switched on.
- inside each interval where the approximation does not change, to integrate the system of cosmological perturbations  $dy[\text{index\_pt}] = \dots y[\text{index\_pt}]$ , using `perturb_initial_conditions()`, `perturb_derivs()`, `perturb_total_stress_energy()`, `perturb_einstein()`
- when the approximation scheme changes, manage to redefine  $y[\text{index\_pt}]$  and ensure continuity
- each times that we cross a value of  $\tau$  where we wish to sample the sources, compute them and store them with `perturbations_sources()`

# Approximation schemes



# Printing the perturbation evolution



- 1 execute `./class` with `k_output_values = 0.01,0.1,1` in the input file
- 2 `python`, `classy`, `cosmo.set({'k_output_values':'0.01,0.1,1', ...})` and `cosmo.get_perturbations()`

Use `test/test_perturbations.c` to print/output only the source functions  $S_X(k, \tau)$ .

## Exercise III

There exist several ways to parametrise modifications of gravity. For instance, people often study the effect of a function  $\mu(k, \tau)$  inserted in the Poisson equation, giving in the synchronous gauge:

$$k^2 \eta - \frac{1}{2} \frac{a'}{a} h' = -\mu(k, \tau) 4\pi G a^2 \bar{\rho}_{\text{tot}} \delta_{\text{tot}} .$$

Localise the above equation and implement, for instance,  $\mu = 1 + a^3$ . Print the evolution of  $\phi$  and  $\psi$  in the standard and modified models, and conclude that the  $C_l^{TT}$ 's should be affected only through the late ISW effect. Get a confirmation by comparing directly the  $C_l$ 's.