LLMs with in-context learning for Algorithmic Theoretical Physics

Anamaria Hell [1 2] **Leander Thiele** [1 2]

Abstract

Theoretical physicists typically spend much time on algorithmic computations. These, while conceptually simple, can nevertheless be time-consuming and contain subtleties that are crucial for the broader research question. Motivated by the recent improvement of Large Language Models (LLMs), we investigate to what degree LLMs equipped with a computer algebra system (CAS) runtime and sufficiently informative context can reliably carry out these algorithmic computations. We task the agent with solving problems from a restricted group, namely cosmological perturbations in modified theories of gravity. We find that worked examples in the context are essential, a small number of representative examples being more effective than a longer context in which solution strategies are duplicated. Conversely, a purely abstract description of the algorithm is insufficient context. Through qualitative analysis we find a strong correlation between the agent's capabilities and the specific strategies available in the context. With an ideal context, we evaluate the agent's ability on par with a first-year PhD student.

1. Introduction

Theoretical physics research consists of a range of tasks of varying complexity. Between the most mechanistic calculations which are relegated to numeric and symbolic computer programs, and the art of constructing new theories born out of genuine creativity, lies a large terrain of "algorithmic computations." By this we mean tasks that are sufficiently complex that a general solution in terms of an explicit computer program is unavailable, while at the same time not being of such difficulty that entirely new methods and ways of thinking are required for their solution.

As algorithmic theoretical physics we define classes of computations for which a generic recipe exists while the details vary from problem to problem and certain subtleties may require special care. Perturbative calculations (like scattering matrices) in renormalizable quantum field theories, string theory, or effective field theories are typical examples. Familiar to general physicists would be boundary value problems in classical electrodynamics or perturbation theory in quantum mechanics.

Such algorithmic computations currently take up much time for theoretical physicists, even with the help of computer algebra systems (CAS). At the same time, their relative simplicity implies that agentic artificial intelligence might solve them reliably. The ubiquity of these problems implies that usually a large set of worked examples is available. Furthermore, an abstract description of the solution strategy can be formulated with varying degrees of concreteness. An agent might learn in-context from examples or abstract descriptions.

We thus test in this work the ability of a frontier large language model, interfaced with a CAS runtime and equipped with an informative context, to solve problems from a specific class of algorithmic computations: *identifying the degrees of freedom for cosmological backgrounds in various theories of gravity*. Cosmological perturbations, small ripples in spacetime and matter, give rise to the structure of the Universe that we observe today (Starobinsky, 1979; Mukhanov and Chibisov, 1981; Chibisov and Mukhanov, 1982; Kodama and Sasaki, 1984; Sasaki, 1986; Mukhanov *et al.*, 1992). They represent the degrees of freedom, the building blocks of theories, which determine their stability and observational consequences. Identifying them is thus the starting point for any further research on the theory's predictions. This essential step can be non-trivial when departing from Einstein's General Relativity.

For the purposes of a controlled study, algorithmic computations of cosmological perturbations have some distinct advantages. It is easy to construct test problems with research-grade complexity and with such obscure features that they are impossible to have been encountered in the LLMs training set. We thus avoid the issue of training contamination.

[1]Center for Data-Driven Discovery, Kavli IPMU (WPI), UTIAS, The University of Tokyo, Kashiwa, Chiba 277-8583, Japan [2]Kavli IPMU (WPI), UTIAS, The University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba 277-8583, Japan. Correspondence to: Anamaria Hell <anamaria.hell@ipmu.jp>.

Proceedings of the AI4Physics Workshop at the 43rd International Conference on Machine Learning (AI4Physics@ICML 2026), Seoul, South Korea. 2026. Copyright 2026 by the author(s).

In fact, in our evaluation we sometimes observe the agent attempting to relate the task to a known problem from the literature. At the same time, the problems have unique and easily checked solutions.

Our work aims to establish how much theoretical physicists can benefit from automated computation specifically in algorithmic problems. To this end, we answer the following questions:

1. *What are current capabilities of frontier LLMs equipped with a CAS runtime;*
2. *How should the LLM's context be used;*
3. *What are typical failure modes and how can they be mitigated.*

Thanks to the restricted set of problems and our available expert knowledge, we can understand the agent's solution strategies beyond simply checking the final answer, thus enabling us to augment the quantitative with a qualitative evaluation. A central aim of this work is to develop practical guidelines for how practitioners can employ their existing worked examples as useful context for agents to autonomously tackle similar problems.

As we demonstrate in Sec. 2, most work on LLMs equipped with a symbolic computation tool has concentrated on mathematics. Compared to these works, we focus less on the model's creativity in finding proof strategies and more on its ability to perform relatively routine computations that are simply burdensome for the human physicist. Note, however, that these computations are not necessarily easy, and the use of a CAS does not guarantee correctness. An interesting aspect of this work is the difficulty in defining a locally relevant context: some manipulations need to be carried out with later steps in mind (for example, solution of background equations may produce roots if done naively, which would lead to problems in subsequent steps). Thus, algorithmic theoretical physics is an interesting testing ground for long contexts.

The remainder of this paper is structured as follows. In Sec. 2, we discuss previous works applying agents equipped with CAS to mathematics and physics problems. In Sec. 3, we describe our setup, experiments, and evaluation strategy. In Sec. 4, we summarize our results both qualitatively and quantitatively. We conclude in Sec. 5.

For simplicity and readability, we use anthropomorphising language to describe the samples we observe from the conditional probability distribution that the language model equals. Similarly, the word "agent" always refers to an artificial agent.

2. Related Work

Using structure in an LLM's response goes back to chain-of-thought (Wei *et al.*, 2023). A natural extension of this idea is to interface the model with tools (e.g., Gao *et al.*, 2023; Chen *et al.*, 2023). Such tool-aided reasoning has found widespread application in mathematics, thanks to the strict verifiability afforded by programs such as Lean (e.g., Gou *et al.*, 2024; Khaitan and Ganesh, 2025; Breen *et al.*, 2025; Nezhad *et al.*, 2026; Luo *et al.*, 2026). Not surprisingly, tool use is found to be especially helpful for more difficult problems (Das *et al.*, 2024). In a similar fashion to this work, in-context learning (e.g., Dong *et al.*, 2024) has been studied as a way to improve the agent's performance in math problems (e.g., Liu *et al.*, 2024; Sultan *et al.*, 2026; Gao *et al.*, 2026). In contrast to explicitly fine-tuned models, general purpose models with useful context are easier to use in a human-AI collaboration and can be more easily adapted if features in the tools (e.g., Lean strategies) change (Breen *et al.*, 2025).

In the context of theoretical physics, using agents for automated problem solving has so far been explored less than in mathematics. Part of the reason may be the difficulty in finding research-grade but still easily gradeable problems; recently there have been efforts towards constructing such benchmarks (Chung *et al.*, 2025; Pan *et al.*, 2026) and using them to evaluate test-time scaling (Gao *et al.*, 2025). These works find that without dedicated tools the current frontier models are unable to solve most research-grade benchmark problems. A recent paper that is similar in spirit to our work is Menzo *et al.* (2026): they let an agent use a specialized Mathematica package to compute Feynman diagrams. In contrast to our work, their agent operates in a much more constrained environment, the main difficulty being keeping track of (binary) conventions. Our work is focused on problems where the agent needs to traverse a higher-dimensional decision space to come up with calculation strategies and deal with subtleties. Much more ambitiously, some recent works (e.g., Agrawal *et al.*, 2026) attempt to let agents perform the entire pipeline of theoretical physics research. Others argue that this approach is unlikely to be successful (Lu *et al.*, 2026). We do not aim to replace the human researcher entirely and thus restrict the experiments to algorithmic theoretical physics problems. Similarly to Lu *et al.* (2026), we anticipate that equipping the agent with a CAS is likely useful.

One of the main aims of this work is to establish how well a current frontier model can use examples or other information provided in the context. Indeed, it has been noted that long contexts can degrade performance (e.g., Li *et al.*, 2024; Liu *et al.*, 2023; Jin *et al.*, 2024; Du *et al.*, 2025; Huang *et al.*, 2024), even specifically for mathematics (Liu *et al.*, 2024). These findings suggest that a small set of well-

chosen example problems (for how to choose, e.g., Yang *et al.*, 2024) should optimize performance. On the other hand, Zou *et al.* (2025) find a mixed picture and Agarwal *et al.* (2024) note that more capable modern frontier models are able to better deal with larger contexts.

3. Methods

In this section, we give a pedagogical introduction to the considered theoretical physics problems (Sec. 3.1), describe our implementation of an LLM equipped with a CAS runtime (Sec. 3.2), list the test problems (Sec. 3.3), describe our quantitative evaluation method (Sec. 3.4), and finally describe the different choices of context we provide to the agent (Sec. 3.5).

3.1. The basics of degrees of freedom

In general, the number of degrees of freedom of a physical system is defined as half the number of initial conditions that are necessary to completely determine the initial-value problem. For example, a free particle in 1+1 dimensions is described by the Lagrangian $L = \frac{1}{2}\dot{q}^2$ where dot denotes the derivative with respect to time, and q the generalized coordinate. By varying the corresponding action $S = \int dt L$ with respect to the generalized coordinate, we find the corresponding equation of motion $\ddot{q} = 0$. To completely determine the initial-value problem, we need to provide two initial conditions – the starting position and velocity $q(t_0)$ and $\dot{q}(t_0)$ – which thus gives one degree of freedom. In quantum field theories and theories of gravity one more often studies the dynamics of fields. In contrast to a free particle, such objects also depend on the point in space, a prime example being the scalar field $\phi(\vec{x}, t)$. We thus conventionally refer to the number of degrees of freedom per point in space.

In theories of gravity the main quantity in addition to matter is the metric tensor, $g_{\mu\nu}$. In flat and cosmological backgrounds, the space-time has spherical symmetry which gives rise to three well-known types of perturbations that we classify according to the group of spatial rotations. The perturbations in the metric tensor can thus be decomposed into scalar, vector and tensor modes. However, not all of these modes will be physical degrees of freedom. For example, general relativity only has two degrees of freedom among the ten components of the metric tensor. In this work, we focus only on actions that are quadratic in the perturbations around particular backgrounds based on the Lagrangian formulation, and determine the physical modes according to the *recipe* described in (Hell *et al.*, 2026). Other approaches have been developed with the purpose of identifying the degrees of freedom, including the Hamiltonian analysis with a Dirac-Bergmann algorithm (Bergmann, 1949; Dirac, 1950; Anderson and Bergmann, 1951), and a *Faddeev-Jackiw* approach (Faddeev and Jackiw, 1988), as well as Lagrangian formalism (see Hell *et al.* (2026) for a general prescription, and references therein), and a more rigorously mathematical approach founded directly in the equations of motion (see Heisenberg (2025) and references therein).

3.2. Implementation

Our experimental setup is relatively simple: the context and problem formulation are input via the LLM's API, which produces CAS commands that are locally executed in the runtime read-eval-print loop (REPL), CAS results are appended to the message history and sent back via the API, and so forth until the agent decides a solution has been found (or aborts). We choose to work with a single agent. We call the API directly instead of relying on a ready-made SDK (e.g., Claude Code) since constant and reproducible model capability is important for our experiments. We choose to work with a fixed, typically rather long, context for multiple reasons:

- It enables us to systematically study the impact of different materials in the context.
- Frontier models are getting increasingly better with long contexts and have reached impressive capabilities already (Agarwal *et al.*, 2024).
- Given the specialized nature of the algorithmic computations we are aiming at, we anticipate practitioners to want a simple prescription for how to apply automated computation for their own work which would usually rely on private example calculations.
- The considered computations can benefit, despite their algorithmic linear nature, from a "look-ahead", as discussed in the introduction.

For the LLM, we use Claude Opus 4-6, and for the CAS we use Maple. Maple is a widely adopted CAS whose syntax the LLM knows well. It is important to turn the human-oriented printing off on Maple (`prettyprint=0`). We find that the agent relatively frequently restarts to a fresh Maple runtime when it notices that it is stuck; we include analysis of these restarts in our quantitative evaluation below. Most of our results are obtained with a limit on the output generation of `max_tokens=1024` and thinking turned off; we discuss experiments with thinking turned on in Sec. 4.5. We run each problem only once because we observe relatively stable behavior when repeating some runs. To control runtime, we set a maximum limit of 100 turns at which point the REPL is aborted. The agent is not informed of this turns limit. Empirically, we find that the turns limit interrupts the loop in situations that do not seem to be leading to success.

3.3. Problems and restrictions

We list the nine test problems in Appendix A. In summary, the first problem (R2Fs) is the well-known R^2 gravity, which should be possible to solve for students on a PhD level. It corresponds to the Starobinsky inflation, which is one of the most well-known models in cosmology, and serves as a reference point. Apart from this model, all remaining problems are novel and have not been previously studied in the literature. The first three problems (sRFs, sRFv, sRFt) focus on cosmological perturbations for scalars, vectors and tensors in a model belonging to the recently introduced framework of constrained gravity (Hell and Sasaki, 2025). Due to its complexity, these problems could easily pose a non-trivial challenge for PhD students or early-career postdocs. These problems are followed by a similar study on flat space-time background with three problems (sRMs, sRMt, sRi2Ms) corresponding to scalar, vector and tensor perturbations. While significantly easier, this set of problems contains subtleties such as non-trivial branching solutions that might be challenging to notice. Finally, the last two problems (sRi2Fs, sRi2Ft) contain a different model with higher derivatives, testing the knowledge of handling such systems which may be still on a PhD or postdoc level.

In creating these problems, we focus only on tensor, or scalar-tensor theories in flat and cosmological backgrounds. A more comprehensive study including also vector, tensor and p-form fields will be presented in follow-up work. Aside from the first problem (R2Fs), the remaining problems are built on the recently introduced constrained scalar-field framework[1] (Hell and Sasaki, 2025), ensuring that their analysis does not appear in the literature. The problems also test the agent on a wide range of subtleties, commonly found in studying the perturbations, such as finding special background solutions that give rise to multiple branches, resolving the constraints, dealing with higher derivatives and reducing their order, as well as analyzing different types of the perturbations.

3.4. Output evaluation

We base our evaluation of the results on a *pass/fail* basis. The solutions to the problems can be divided into four main steps: the setup, computing the background equations of motion, finding the second-order action, and analyzing the perturbations. Theories that also involve higher-derivative terms have an additional step in the reduction of the derivatives that can happen either in the setup or at the level of perturbations.

Mirroring the solution strategy, we adopt the following rules for the evaluation:

1. If the setup is wrong, it is a fail.
2. If the background equations of motion are not computed or not properly substituted in the action, and this affects the result, then it is a fail. Otherwise, if this does not change the result, it is a pass.
3. The perturbations and the reduction of the higher-order derivatives should be analyzed correctly for a pass.
4. If the resulting code obtains a correct final result, but forgets a subtlety, then it is still a pass.

3.5. Contexts

In the spirit of in-context learning, we supplement our Claude-Maple interface with four choices of context:

Context 10ex: A set of 10 solved examples, 60k tokens. We prepare 10 step-by-step solved and explained examples of finding propagating degrees of freedom of theories of gravity in cosmology and flat space-time. A short summary of each example is given in Appendix B. Most of these examples are just standard examples, meant to cover different simplifications and Maple code when studying scalar, vector or tensor perturbations. They do not involve any higher derivatives.

Context 3broad: A set of 3 solved examples, 18k tokens. We test the case when the number of examples is overall reduced to three. To still keep it general enough, we select three representative examples (2, 3, and 8) from the 10ex context. We also add the instruction Note that if the problem said "flat background", you should assume Minkowski space-time, and set a(t)=1 too. as preliminary experiments without this instruction lead to frequent misinterpretation of the test problems.

Context 3tailored: A modified set of 3 solved examples, 24k tokens. As we will see in the results, attempts to some problems with contexts 10ex and 3broad fail with relatively common mistakes. To examine this further, we keep the Example 3 and Example 8 as part of the input data, and supplement the two with another modified example that is supposed to account for the common mistakes, given in Sec. 4.3. The same clarification about flat backgrounds as in 3broad is added. Finally, in the previous sets, we notice that the agent spends many turns manipulating the form of the final action. To reduce the number of turns, we add an additional instruction: If you have reached the final action, do not simplify it further.

Context instruction: A general description of the method, 2k tokens. Finally, we provide as an input just a general instruction text based on the recent work (Hell *et al.*, 2026), and with no example code. This *recipe* is given in the Appendix C.

[1] See also (Hell and Daniel, 2026) for a theory with constrained vector field.

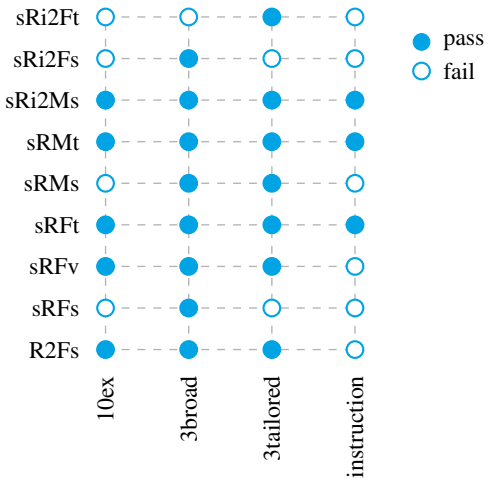

Figure 1. Results for the problems (vertical axis) depending on the provided context (horizontal axis).

We note that the context 3tailored implies some data leakage as it is constructed in response to results from 10ex. This is deliberate: we aim to qualitatively evaluate the correlation between context and test performance. While the 3broad context is also slightly modified in response to observed behavior, the modification is only a small clarification and thus does not constitute data leakage.

4. Results

In Fig. 1, we visualize the solutions' evaluation according to problem (vertical axis) and context (horizontal axis). As an additional quantitative evaluation, we also provide the number of turns and number of Maple restarts in Fig. 2. The baseline case of a long context (10ex) yields 5 correct solutions and 4 incorrect ones. For the shorter context of representative problems (3broad), the success rate improves dramatically to 8 out of 9 (with a caveat discussed in Sec. 4.2 below). The case of specifically tailored problems (3tailored), which is informed by the previous two experiments, shows a similarly high success rate of 7 out of 9 and is the only one where sRi2Ft is solved correctly. The 3tailored experiments also display the lowest number of turns and restarts on average, indicating that the deliberately constructed examples help the model find the solution efficiently. The experiments with an abstract description of the *recipe* (instruction) show the lowest number of successes and the highest average number of turns and restarts. For a given problem, a higher number of turns and restarts is correlated with lower success rate.

In the following, we discuss in detail the results with each of the four contexts, followed by a brief comparison of thinking mode and reduced temperature for selected problems.

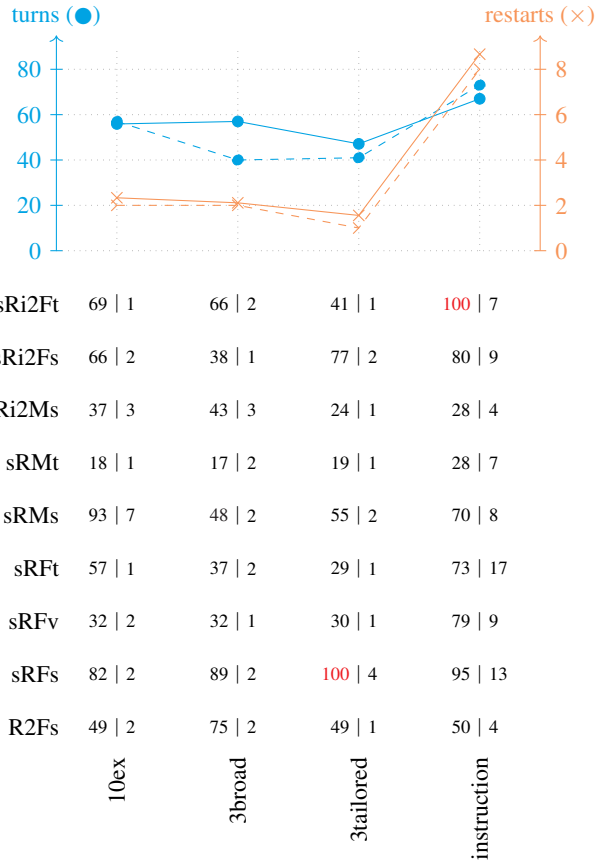

Figure 2. The lower panel shows the solution length as turns|restarts depending on context (horizontal axis) and problem (vertical axis). The upper panel summarizes the table in terms of mean (solid) and median (dashed).

4.1. Context 10ex

The agent solves five problems correctly out of nine. The number of turns – sequences in which the agent runs Maple – ranges from the minimal 18 to 93, with an average of 56, while the number of restarts[2] is mostly from 1 to 3, with a special exception of 7 restarts in sRMs, and has an average of 2.

Among the given problems, the well-known Starobinsky inflation model R2Fs is a very useful reference in confirming that the agent uses not only the examples submitted in the context but also information fed during training. In particular, while its task is to utilize the Jordan frame,[3] it still introduces an external scalar field, rewriting the initial action instead as $\int \sqrt{-g}\left(\frac{M_P^2}{2}+2\beta\sigma\right)R-\beta\sigma^2$ which re-

[2]The minimal number of restarts is one, corresponding to the start of the code.

[3]This particular problem is conventionally solved in the Einstein frame so we aimed to reduce reliance on training data by restricting the solution strategy.

duces to the original theory by resolving the constraint for σ, and substituting it back to the action. This significantly simplifies the analysis, as it removes all of the higher derivatives acting on the perturbations that would otherwise be present until the order is reduced.

Throughout the remaining problems, one of the major challenges is the presence of higher-order derivatives. The agent mostly deals with them through an incorrect substitution, with the aim of reducing the order of the higher derivatives acting on the perturbations. In particular, the main issue is that the agents forgets to include the constraints via the Lagrange multipliers, and thus reduces the order incorrectly. This is the case for sRFs, sRi2Fs and sRi2Ft.

Another challenge is resolving the background equations of motion. The above test problems are constructed with the intention of having multiple branching solutions, and thus challenging the agent in this way as well. Surprisingly, however, different branches are noticed for all but sRMt, with the agent then selecting one of the branches (usually the more complicated one). The main issue associated to the background equations of motion is to resolve both constraint equation, acceleration equation and the equation for the scalar in such a way that there are no roots appearing in the solutions. In other words, the challenge is how to pick the variables which would be best to solve the equations for, when substituting them back to the action. Interestingly, in sRFv and sRFt, the agent also struggles with the background equations of motion, gives up and proceeds to the second-order action. Only once there, it returns and studies the background equations, analyzing if it can simplify the resulting action at that point.

A noteworthy case is sRMs. The agent first correctly solves for the flat background. However, then, it deems the result too trivial, and instead turns to the analysis in curved spacetime.

4.2. Context 3broad

The number of solved problems is 8 out of 9. The average number of turns is 49 and the average number of restarts is 2.

R2Fs is solved in a very similar way as in the previous case. Interestingly, the agent then uses the same trick to also solve sRFs, rewriting the action of the problem as

$$S = \int d^4x\sqrt{-g}\left[\left(\frac{1}{2}\sigma^2 f'(\chi) + \frac{M_P^2}{2}\right)R - M_P^2\Lambda + \frac{1}{2}\sigma^2\left(m^2 + f(\chi) - \chi f'(\chi)\right)\right].$$

Strictly speaking, this action is equivalent to the previous one only if $f''(\chi) \neq 0$. However, aside from this subtlety, the problem is now solved correctly, unlike with context 10ex. Another correctly solved problem is sRi2Fs. While the agent notices different branching solutions depending on whether the constrained scalar vanishes or not, it chooses the simpler one with $\sigma(t) = 0$. This procedure is as valid as choosing the arbitrary branch, although it simplifies the analysis significantly. In addition, sRFv and sRFt are mostly correctly solved. However, we note that background equations of motion were overall unsolved or ignored. Since this still does not affect the final expression too much, we grade it with a passing grade, but one should keep in mind that this could have easily led to the wrong result in more complicated scenarios.

For problems sRMs, sRMt, and sRi2Ms, we find that all are correctly solved according to our evaluation criteria. However, we note that a big part in this is the choice of an easier branch of solutions, in which the background field vanishes, whereas with other contexts the agent usually chooses the more complicated, non-vanishing branch.

The remaining completely unsolved problem is sRi2Ft for which, just as in 10ex, the presence of higher-order derivatives proves too much for the agent.

4.3. Context 3tailored

The primary issues in the previous two contexts are

- reducing the higher derivatives at the level of perturbations,
- solving background equations of motion.

To counter these issues, in this context we replace the Example 2 with the following example on the pure R^2 gravity:

Modified example: Consider βR^2 on a cosmological background, where β is the coupling constant. Find the quadratic action for the scalar perturbation around this background in the Jordan frame, and express it in terms of the propagating modes.

This problem inherently contains the higher derivatives in the Jordan frame, which are reduced on the level of perturbations to show that the theory describes one scalar mode on cosmological backgrounds (in addition to the two tensor ones), with the aim of countering the first issue found in the previous analyses.

In addition, we add to the context further explanations and tips on how to solve the background equations of motion in more general cases.

Overall, the performance of this set is significantly improved compared to the 10ex context but not compared to 3broad. Out of nine problems, two were not solved. For sRFs, the agent reaches 100 turns and is terminated. It also does not perturb the constrained scalar in addition to the metric per-

turbations. The latter issue is also present in the sRi2Fs problem, which is otherwise very close to the correct solution. In contrast to the previous contexts this context does not include an example where an external scalar field (such as the constrained scalar) is perturbed, but only contains metric (scalar) perturbations. Potentially, therefore, one might suspect that by adding an example where this is covered, one would find all of the problems solved.

On average, the number of turns is 47, while the average number of restarts is 2, with the maximum of 4 in the sRFs. Notably, in contrast to previous cases, R2Fs, which is well-known in the literature, is now solved purely in the Jordan frame, with no introduction of another scalar at the level of action, but rather by reducing the higher derivatives on the level of perturbations, as the context example shows.

4.4. Context instruction

This context does not include any code, but instead describes an algorithm to compute the number of degrees of freedom for the given problems. As a result, the agent does not analyze any of the problems by fully formulating them in Maple and running there the analysis. Instead, for some of the problems, it combined the analytical reasoning "by hand", and a bit of Maple code, up to expanding the Ricci scalar or Ricci tensor in the action, to infer the final form of the perturbations.

On average, the number of restarts per run is 9, with a maximum of 17 in sRFt, and a minimum of 4 in R2Fs and sRi2Ms. The number of turns is significantly higher than the rest of the cases, averaging 67.

Overall, its performance is significantly lower than the other cases, with only sRFt, sRMt and sRi2Ms getting a passing grade. What these problems have in common is the actions are particularly simple, as the agent evaluates the tensor modes and scalars on flat background, the latter taken with vanishing background value of the scalar field. However, we note that these problems are not solved perfectly. The agent never computes the full set of background equations of motion. In addition, in sRFt, the action given in the summary of the computation differs by a (correct) factor of $1/2$ from the action that was previously derived. In the cases of flat background, the agent understands it as a cosmological background, as is the case for context 10ex.

The largest issue among the failed problems is the decision-making. For example, in sRFs, the agent never decides on the approach, and never performs any computations in Maple. In fact, the trend of doing minimal Maple computations extends to many other problems, resulting in a fail. Among them, in particular, sRi2Ft reaches the turns limit, while for sRi2Fs the agent simply gives up after 80 turns.

4.5. Thinking enabled

The previous main results are obtained with thinking turned off. In the following, we present results with "medium" effort thinking and increased `max_tokens=8192`.

Our analysis mainly focuses on re-analysing the 10ex context. The performance with medium thinking is noticeably improved when compared to the baseline case. The average number of turns is reduced to 40. The mean number of restarts is 1.9 as compared to 2.3. Thus we overall find that the analysis is faster and more focused.

Notably, two cases switch from fail to pass, whereas the rest keeps similar results as without thinking.

The case sRFs fails in the baseline due to the incorrect handling of the higher derivatives. With thinking, the agent notices the presence of the higher derivatives, and then restarts and rewrites the starting action with the help of an external scalar field which reduces their order. Therefore, it handled this obstacle in a way similar to the set 3broad (without thinking), leading to a correct result.

Another case with significant improvement is sRMs. Originally, this case is failed because the agent concludes that the result is too trivial after finding no propagating scalar modes, evaluated on the branch in which the constrained scalar vanishes, and then reformulates the problem into the one with cosmological background. Here, the agent performs exactly the same procedure, and states that the issue might be too trivial. However, instead of switching the background to the cosmological one, it rather utilizes the constraint equation for the constrained scalar and studies the other branch of solution, which leads it to the correct result.

In addition to the 10ex set, we analyse representative problems in the other sets that are failed without thinking:

3broad — sRi2Ft. The agent correctly notices that there are higher-derivative terms. Notably, this is the single case where it presents the results in terms of the higher-order action. However, part of its computation is incorrect: after noticing the higher derivatives, it reduces their order in an incorrect way, which leads it to the conclusion that the theory propagates fewer degrees of freedom. After this, it suddenly switches and writes the final result in terms of the higher-order action with a correct conclusion. It is interesting that this sudden switch was not explained in the output, and did not happen in any other runs for the other problems.

3tailored — sRi2Fs. In contrast to the previous run, with thinking the agent finishs within 40 turns without restarting. However, it makes exactly the same mistake as in the initial run, forgetting to perturb the constrained scalar in addition to the metric perturbations. Therefore, the result still remains a fail.

instruction — sRi2Fs. In contrast to the previous problems, the agent does not primarily use Maple in this analysis even with the thinking mode, apart from some simple computations, and gives a wrong result.

4.6. Temperature

The previous results use the default temperature $T = 1$. Since algorithmic workflows are intuitively expected to benefit from lower temperatures, in this section we present results with $T = 0.2$. We note that according to documentation, current Claude versions are not supposed to be evaluated with non-default temperature and thus we present the default as the main results.

We test the reduced temperature with the 10ex context, evaluating all of the test problems. All problems apart from sRFs, sRi2Fs and sRi2Ft are solved correctly. Compared to the $T = 1$ case, the average number of turns is reduced to 46, the sRFs attempt being terminated at 100 turns, while the average number of restarts is 2.

Similarly to the default setting, the agent has difficulty solving problems with higher derivatives. While it recognizes that one way to reduce the action is by reducing the derivatives, when it introduces additional fields, it forgets the Lagrange constraint. Therefore, while it claims the correct result, due to this mistake in the procedure we note this as a failed exercise.

5. Conclusions

In their daily work, theoretical physicists encounter many problems of algorithmic character. We explore how well a frontier LLM equipped with a CAS runtime can solve such problems. Specifically, we focus on finding the degrees of freedom in modified theories of gravity and utilize completely novel problems for evaluation.

Equipped with qualitative and quantitative evaluations described above, we can now answer the questions posed in the introduction as follows:

1. *What are current capabilities of frontier LLMs equipped with a CAS runtime?* When supplied with worked examples, the agent is able to competently use the CAS REPL and in a majority of cases solves the test problems correctly. Whenever particular complications are not included in the example context, the agent often makes mistakes. We evaluate the agent's ability on par with a first-year graduate student.

2. *How should the LLM's context be used?* We find that worked examples are essential for the agent to succeed whereas an abstract description of the algorithm is not sufficient. Practitioners will typically have worked examples available from their prior work; we recommend to use such examples for in-context learning. We find that a smaller set of example problems, chosen for representativeness, is more effective than a large set. A small, targeted set can mitigate observed failure modes but may introduce new issues. Through our qualitative evaluation we observe a strong correspondence between the context and the agent's performance (a particularly striking example being the higher derivatives), implying that the worked examples are not merely used for CAS syntax.

3. *What are typical failure modes and how can they be mitigated?* One common failure is misinterpretation of the problem (typically, because the solution of the actual problem is deemed "trivial" – not unlike students in exams). A human in the loop would be able to correct this type of problem. Other mistakes such as failure to correctly apply the background equations are more worrisome but can be caught by an experienced supervisor. We find that tailoring the context with handcrafted examples that cover observed failure modes can be effective in reducing mistakes.

While the recommendations to choose the context deliberately and utilize human-AI collaboration are expected, we also make some unexpected findings. The test problems are tailored to contain subtleties such as branching solutions. Our expectation that their existence would be overlooked was wrong: the agent identifies most of the special cases correctly and uses some of them to simplify the analysis. We had also expected that enabling thinking would be more useful, especially for the cases with an abstract description of the algorithm in the context. While thinking does improve performance at the margins, the general picture that examples in the context are essential remains unchanged.

Two aspects of the agent's behavior are noteworthy. First, it shows strong perseverance and goal-orientedness. In several examples we observe its pursuing a wrong path, noticing that it is getting nowhere, and then `restart`ing the Maple runtime for a new attempt. Second, the agent shows a good ability to consider the problems abstractly. Sometimes an obstacle prevents it from correctly solving the background equations of motion, at which point it proceeds with whatever it can accomplish in the rest of the computation while keeping the background solution as an abstract object. Later it tries to go back and obtain the concrete solution.

We have only explored one class of algorithmic theoretical physics problems, with a single frontier LLM, a single CAS, and a fixed context. Future work should pursue the following directions:

- generalization across a broader spectrum of algorithmic theoretical physics computations,

- comparison beetween models and with other CAS, such as Mathematica,
- RAG-style setup for locally more relevant dynamic context (either using human-created labels of parts in the worked examples, or using an automated preprocessing step of the available examples),
- adaptive compactification of long algebraic expressions to improve information density in the context,
- asynchronous interaction with the CAS, similarly to how humans explore other strategies or work on different parts of the problem while a symbolic manipulation is taking a long time.

We expect these improvements to be rather useful for the LLM-CAS agent. The final two research directions could also provide impulses for the broader field of agentic science.

As we observe strong performance on the considered test problems, we believe that CAS-equipped agents with in-context learning will be useful for other algorithmic computations across string theory, quantum field theory, gravity, and cosmology. At the same time, we observe failures that only close supervision by experienced researchers would catch, and thus recommend particular care in curating the example context as well as interactive human-AI collaboration.

Acknowledgements

AH would like to thank Hugo Allaire, Elisa Ferreira, Jia Liu, Misao Sasaki, Tadayuki Takahashi, and Ying-Li Zhang for useful discussions. LT would like to thank Fatih Dinç for advice. AH is supported by JSPS KAKENHI Grant No. JP26K17133, and by the CD3 Google Seed grant. LT is supported by JSPS under KAKENHI 24K22878 and 26K17136 and by the Royal Society under ICA\R2\252140.

References

A. A. Starobinsky, JETP Lett. **30**, 682 (1979).

V. F. Mukhanov and G. V. Chibisov, JETP Lett. **33**, 532 (1981).

G. V. Chibisov and V. F. Mukhanov, Mon. Not. Roy. Astron. Soc. **200**, 535 (1982).

H. Kodama and M. Sasaki, Prog. Theor. Phys. Suppl. **78**, 1 (1984).

M. Sasaki, Prog. Theor. Phys. **76**, 1036 (1986).

V. F. Mukhanov, H. A. Feldman, and R. H. Brandenberger, Phys. Rept. **215**, 203 (1992).

J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, Chain-of-thought prompting elicits reasoning in large language models (2023), arXiv:2201.11903 [cs.CL] .

L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig, Pal: Program-aided language models (2023), arXiv:2211.10435 [cs.CL] .

W. Chen, X. Ma, X. Wang, and W. W. Cohen, Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks (2023), arXiv:2211.12588 [cs.CL] .

Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, M. Huang, N. Duan, and W. Chen, Tora: A tool-integrated reasoning agent for mathematical problem solving (2024), arXiv:2309.17452 [cs.CL] .

A. Khaitan and V. Ganesh, O-forge: An llm + computer algebra framework for asymptotic analysis (2025), arXiv:2510.12350 [cs.AI] .

B. Breen, M. D. Tredici, J. McCarran, J. A. Mijares, W. W. Yin, K. Sulimany, J. M. Taylor, F. H. L. Koppens, and D. Englund, Ax-prover: A deep reasoning agentic framework for theorem proving in mathematics and quantum physics (2025), arXiv:2510.12787 [cs.AI] .

S. B. Nezhad, Y. Li, and A. Agrawal, Symcode: A neurosymbolic approach to mathematical reasoning via verifiable code generation (2026), arXiv:2510.25975 [cs.CL] .

H. Luo, H. Feng, Q. Sun, C. Xu, K. Zheng, Y. Wang, T. Yang, H. Hu, and Y. Tang, Agentmath: Empowering mathematical reasoning for large language models via tool-augmented agent (2026), arXiv:2512.20745 [cs.AI] .

D. Das, D. Banerjee, S. Aditya, and A. Kulkarni, Mathsensei: A tool-augmented large language model for mathematical reasoning (2024), arXiv:2402.17231 [cs.CL] .

Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, B. Chang, X. Sun, L. Li, and Z. Sui, A survey on in-context learning (2024), arXiv:2301.00234 [cs.CL] .

J. Liu, Z. Huang, C. Wang, X. Huang, C. Zhai, and E. Chen, What makes in-context learning effective for mathematical reasoning: A theoretical analysis (2024), arXiv:2412.12157 [cs.CL] .

O. Sultan, E. Stern, and D. Shahaf, A neuro-symbolic approach for reliable proof generation with llms: A case study in euclidean geometry (2026), arXiv:2505.14479 [cs.AI] .

A. Gao, C. Zhang, X. Zhang, D. Li, M. Zhao, F. Liu, and X. Zhang, Process in-context learning: Enhancing mathematical reasoning via dynamic demonstration insertion (2026), arXiv:2601.11979 [cs.AI] .

D. J. H. Chung, Z. Gao, Y. Kvasiuk, T. Li, M. Münchmeyer, M. Rudolph, F. Sala, and S. C. Tadepalli, Theoretical physics benchmark (tpbench) – a dataset and study of ai reasoning capabilities in theoretical physics (2025), arXiv:2502.15815 [cs.LG] .

H. Pan, J. V. Roggeveen, E. Berg, J. Carrasquilla, D. Chowdhury, S. Ganguli, F. Ghimenti, J. Hasik, H. Hunt, H.-C. Jiang, M. Kamb, Y.-J. Kao, E. Khatami, M. J. Lawler, D. Luo, T. Neupert, X. Qi, M. P. Brenner, and E.-A. Kim, Cmt-benchmark: A benchmark for condensed matter theory built by expert researchers (2026), arXiv:2510.05228 [cs.LG] .

Z. Gao, T. Li, Y. Kvasiuk, S. C. Tadepalli, M. Rudolph, D. J. H. Chung, F. Sala, and M. Münchmeyer, Test-time scaling techniques in theoretical physics – a comparison of methods on the tpbench dataset (2025), arXiv:2506.20729 [cs.LG] .

T. Menzo, A. Roman, G. T. Fleming, S. Gleyzer, K. T. Matchev, and S. Mrenna, Agentic diagrammatica: Towards autonomous symbolic computation in high energy physics (2026), arXiv:2603.26990 [hep-ph] .

P. Agrawal, N. Craig, A. Madden, and I. V. Lombera, The fermiacc: Agents for particle theory (2026), arXiv:2603.22538 [hep-ph] .

S. Lu, Z. Jin, T. J. Zhang, P. Kos, J. I. Cirac, and B. Schölkopf, Can theoretical physics research benefit from language agents? (2026), arXiv:2506.06214 [cs.CL] .

T. Li, G. Zhang, Q. D. Do, X. Yue, and W. Chen, Long-context llms struggle with long in-context learning (2024), arXiv:2404.02060 [cs.CL] .

N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, Lost in the middle: How language models use long contexts (2023), arXiv:2307.03172 [cs.CL] .

B. Jin, J. Yoon, J. Han, and S. O. Arik, Long-context llms meet rag: Overcoming challenges for long inputs in rag (2024), arXiv:2410.05983 [cs.CL] .

Y. Du, M. Tian, S. Ronanki, S. Rongali, S. Bodapati, A. Galstyan, A. Wells, R. Schwartz, E. A. Huerta, and H. Peng, Context length alone hurts llm performance despite perfect retrieval (2025), arXiv:2510.05381 [cs.CL] .

X. Huang, L. L. Zhang, K.-T. Cheng, F. Yang, and M. Yang, Fewer is more: Boosting llm reasoning with reinforced context pruning (2024), arXiv:2312.08901 [cs.CL] .

X. Yang, J. Lin, Z. Wang, and C. Zhai, Learning by analogy: Enhancing few-shot prompting for math word problem solving with computational graph-based retrieval (2024), arXiv:2411.16454 [cs.CL] .

K. Zou, M. Khalifa, and L. Wang, On many-shot in-context learning for long-context evaluation (2025), arXiv:2411.07130 [cs.CL] .

R. Agarwal, A. Singh, L. M. Zhang, B. Bohnet, L. Rosias, S. Chan, B. Zhang, A. Anand, Z. Abbas, A. Nova, J. D. Co-Reyes, E. Chu, F. Behbahani, A. Faust, and H. Larochelle, Many-shot in-context learning (2024), arXiv:2404.11018 [cs.LG] .

A. Hell, E. G. M. Ferreira, D. Lüst, and M. Sasaki, JHEP **03**, 235, arXiv:2601.10288 [hep-th] .

P. G. Bergmann, Phys. Rev. **75**, 680 (1949).

P. A. M. Dirac, Can. J. Math. **2**, 129 (1950).

J. L. Anderson and P. G. Bergmann, Phys. Rev. **83**, 1018 (1951).

L. D. Faddeev and R. Jackiw, Phys. Rev. Lett. **60**, 1692 (1988).

L. Heisenberg, (2025), arXiv:2509.18192 [math-ph] .

A. Hell and M. Sasaki, (2025), arXiv:2507.00986 [hep-th] .

A. Hell and T. Daniel, (2026), arXiv:2603.18147 [hep-th] .

A. Test problems

To analyse the usefulness of Maple-equipped Claude in studying the cosmological perturbations, we evaluate it on a set of nine problems, given by the following:

R2Fs Consider $\frac{M_P^2}{2}R + \beta R^2$ gravity on a cosmological background, where M_P is the Planck mass, and β is the coupling. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes. Perform the calculation in the Jordan frame, you are not allowed to use the Einstein frame.

sRFs Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + f(R))$ gravity on a cosmological background, where M_P is the Planck mass, Λ is the cosmological constant, σ is the scalar field, m^2 is the mass, and $f(R)$ is a function of Ricci scalar. Find the quadratic action for the scalar perturbations around this background, and express it in terms of the propagating modes.

sRFv Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + f(R))$ gravity on a cosmological background, where M_P is the Planck mass, Λ is the cosmological constant, σ is the scalar field, m^2 is the mass, and $f(R)$ is a function of Ricci scalar. Find the quadratic action for the vector perturbations around this background, and express it in terms of the propagating modes.

sRFt Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + f(R))$ gravity on a cosmological background, where M_P is the Planck mass, Λ is the cosmological constant, σ is the scalar field, m^2 is the mass, and $f(R)$ is a function of Ricci scalar. Find the quadratic action for the tensor perturbations around this background, and express it in terms of the propagating modes.

sRMs Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + f(R))$ gravity on a flat background, where M_P is the Planck mass, σ is the scalar field, m^2 is the mass, and $f(R)$ is a function of Ricci scalar. Find the quadratic action for the scalar perturbations around this background, and express it in terms of the propagating modes.

sRMt Consider $\frac{M_P^2}{2}R + \frac{1}{2}\sigma^2(m^2 + f(R))$ gravity on a flat background, where M_P is the Planck mass, σ is the scalar field, m^2 is the mass, and $f(R)$ is a function of Ricci scalar. Find the quadratic action for the tensor perturbations around this background, and express it in terms of the propagating modes.

sRi2Ms Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + \xi R_{\mu\nu}R^{\mu\nu})$ gravity on a flat background, where M_P is the Planck mass, σ is the scalar field, m^2 is the mass, and ξ is the coupling constant. Find the quadratic action for the scalar perturbations around this background, and express it in terms of the propagating modes.

sRi2Fs Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + \xi R_{\mu\nu}R^{\mu\nu})$ gravity on a cosmological background, where M_P is the Planck mass, σ is the scalar field, m^2 is the mass, and ξ is the coupling constant. Find the quadratic action for the scalar perturbations around this background, and express it in terms of the propagating modes.

sRi2Ft Consider $\frac{M_P^2}{2}(R - 2\Lambda) + \frac{1}{2}\sigma^2(m^2 + \xi R_{\mu\nu}R^{\mu\nu})$ gravity on a cosmological background, where M_P is the Planck mass, σ is the scalar field, m^2 is the mass, and ξ is the coupling constant. Find the quadratic action for the tensor perturbations around this background, and express it in terms of the propagating modes.

We note that the above problems are passed in plain text, and only set in an appropriate LaTeX form for readability. An actual input looks like the following example:

```
Consider Mp^2/2*R+beta*R^2 gravity on a
cosmological background, where Mp is the
Planck mass, and beta is the coupling.
Find the quadratic action for the scalar
perturbation around this background, and
express it in terms of the propagating
modes.  Perform the calculation in the
Jordan frame, you are not allowed to use
the Einstein frame.
```

B. Presentation of the problems for the context 10ex with one solved example

The full list of the problems for which we have presented detailed code with explanations in Maple is given by the following:

Example 1: Consider the w=0 Brans-Dicke theory with a linear potential on a cosmological background. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes.

Example 2: Consider a canonical scalar field theory with an arbitrary potential on a cosmological background with Einstein gravity. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes.

Example 3: Consider a canonical scalar field theory with an arbitrary potential on a cosmological background with Einstein gravity. Find the quadratic action for the vector perturbation around this background, and express it in terms of the propagating modes.

Example 4: Consider the action consisting of Einstein gravity with cosmological constant, and a massive vector field that is non-minimally coupled to gravity through an Einstein term. Find the tensor modes, in case when the vector field has a vanishing background value, but the space-time

corresponds to the de Sitter Universe.

Example 5: Consider the Einstein gravity in flat space-time. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes.

Example 6: Consider the Einstein gravity with cosmological constant in flat space-time. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes.

Example 7: Consider a non-minimally coupled scalar field to the Ricci scalar, with a sin-potential. Express the action in terms of the propagating scalar modes.

Example 8: Consider Einstein gravity with a cosmological constant, and a scalar field with a quartic potential. Find the quadratic action for the tensor perturbations around this background, and express it in terms of the propagating modes.

Example 9: Consider a k-essence field theory with an arbitrary potential on a cosmological background with Einstein gravity. Find the quadratic action for the vector perturbation around this background, and express it in terms of the propagating modes.

Example 10: Consider a k-essence field theory with an arbitrary potential on a cosmological background with Einstein gravity. Find the quadratic action for the scalar perturbation around this background, and express it in terms of the propagating modes.

One should note that while the above formulations are essentially loose, they are supplemented with a solved step-by-step code, and thus each of them automatically comes with a defined action.

C. The recipe for the context instruction

Work-out the perturbations based on the following general instructions:

The following is VERY IMPORTANT: SET interface(prettyprint = 0):

(A) Specify the background One of the main reasons to study any theory is to describe physical phenomena, such as, for example, the acceleration of the Universe, black-holes, or cosmological defects. These are the solutions to the background equations of motion, found by varying the action with respect to fields. Notably, depending on the background, the number of degrees of freedom (dof) can change. Therefore, when performing the analysis to calculate the dof, it is best to assume that the background is always satisfied. One can ensure this by simply assuming that the background equations of motion always hold at any step in the computation. (B) Prepare the perturbations Once you specify the background (which can also be trivial), it's time to perturb around it. These perturbations will provide the notion of the degrees of freedom. To find them, there are several preparatory things to consider before studying their corresponding action. In particular, it is useful to answer the following questions to decide on the best way forward: (i) Does your system have any symmetries? Here, by symmetries, we mean what is left after one separates the time and space components. If then the answer is yes, this might suggest the form of your perturbations, from a general study of components of your field to a much simpler form. For example, in space-times that are homogeneous and isotropic, it is possible to do a decomposition of the modes according to the group of spatial rotations, and decompose the fields into scalars, vectors, and tensors. If the space-time has broken anisotropies in one direction and three spatial dimensions, one can instead decompose the components of the fields into even and odd modes.

(B) Does your system have any gauge redundancy? If yes, analyzing first how the perturbations are transforming under gauge redundancy could be very beneficial: if one fixes the gauge such that it matches the gauge invariant variables, it can decrease the length of the computation. Other gauges are possible too, as long as one makes sure that the gauge is fully fixed. It should be stressed that the above two questions can be avoided – it is perfectly fine if one considers all components of the fields, and does not even fix the gauge. These two decisions would lead to equivalent results, but, also can make the computation significantly longer.

(C) Expand the action Once one decides on the background and the type of perturbations around it, it is time to study the perturbations themselves. This can be analyzed either on the level of action or directly on the level of equations of motion. However, studying them directly from the action is more beneficial, since one can then have a clear view into the conditions under which the degrees of freedom are well behaved, such as the positivity of the factors multiplying their kinetic terms, necessary to avoid the ghost modes.

In general, one is supposed to expand the action to the required order in perturbations. If there is a non-vanishing background for the fields, the leading order corresponds to the equations of motion, which are subsequently satisfied. Thus, at first order, the corrections vanish. The kinetic terms for the perturbations are then at second order in the action, and one can also study higher orders to analyze the non-linear corrections. Note that this also holds if the background is trivial, eg, including vanishing background values of the fields, together with the Minkowski space-time.

(D) Analyze the theory Once the action is ready, it is time to analyze the modes. Let us first assume that there are at most two time derivatives acting on the fields, and that all

constraints are holonomic. Then, essentially, the procedure to find the propagating modes is given by the following: (i) Locate the fields that are not propagating – the constrained modes. (ii) Find their corresponding constraint by varying the action with respect to them. (iii) Solve it, and substitute it back into the action. Note that in the case of non-linearities, this is usually done perturbatively. (iv) Check if the determinant of the kinetic matrix for the Lagrangian density of fields is vanishing. If yes, this means that there are more non-propagating fields. Locate them in the Lagrangian density. If all of them appear with the form where they carry terms quadratic in time-derivatives, this means that one should perform a substitution, which will render one of them to be non-propagating, and then one can repeat the procedure (i) – (iv). (v) If, after repeating the procedure, the determinant of the kinetic matrix is not vanishing, and your system contains no higher derivatives, then you have found the Lagrangian density in terms of the propagating modes.

(E) Higher-order derivatives The above procedure works for a wide range of theories. However, one can also easily encounter theories that contain higher-order derivatives. Luckily, this does not drastically complicate the analysis. The key to analyzing it is to reduce the order of derivatives until one reaches maximally two time derivatives per term in the Lagrangian.