Imagination and Planning*

Gabrielle Gayer†and Itzhak Gilboa‡

January 2025

Abstract

We consider a model of case-based planning, where a *position* is a vector of numbers, and a *case* is an edge in the directed graph of positions. The planner generates new plans by using cases that are similar to those she has observed in the past. In the benchmark model presented here, similarity is defined by equality of differences (between the target and the source position). We prove a complexity result that shows why planning requires imagination and is not easily done algorithmically. We put this result in the context of learning and expertise in case-based models, distinguishing among information, insight, and imagination.

1 Introduction

1.1 Motivating Examples

This note suggests a simple model of case-based planning. The basic idea is that plans are constructed from cases, each of which is similar to a case that has been encountered in the past, though the particular concatenation of these

*We thank Thomas Astebro and Carlos Serrano for discussions that motivated us to revisit this project. We also thank Mateus Hiro Nagata and Nitin Yadav, and especially to Scott Boorman for comments and references. We are particularly grateful Gilboa gratefully acknowledges Investissements d'Avenir ANR -11- IDEX-0003 / Labex ECODEC No. ANR - 11-LABX-0047, the AXA Chair for Decision Sciences at HEC.

†Bar-Ilan University. gabi.gayer@gmail.com

‡HEC, Paris, and Reichman University. tzachigilboa@gmail.com

cases may be new, and specific to the current plan.[1] The following examples illustrate.

Example 1: Anne is in Paris and she wants to get to Hanoi. She heard that a colleague of hers, Jean, flew from Paris to Bangkok, and then took a flight from Bangkok to Melbourne. She also knows that her friend, Patricia, flew from San Francisco to Bangkok and from there to Hanoi. Anne devises a plan to fly from Paris to Bangkok, as did Jean, and from there to Hanoi, as did Patricia. She does not know of anyone who followed this particular route, but, for each segment thereof, she knows of a past case indicating that this segment is possible (or at least was possible in the past). She uses her imagination to use these past cases to generate a new plan. (See Figure 1.)

Example 2: Claire has an idea for a startup. She needs \$1M in seed money to hire programmers, and then develop a product, or at least a proof of concept. She knows of Mike, who obtained \$1M from a VC fund. Mike, who already had a team of programmers and did not need any more, used the seed money for other purposes. Claire also knows of Diane who owns a firm that recently hired programmers with an investment of \$1M. Diane's firm is well-established and it started out with \$10M in available funds. Claire comes up with a plan, to obtain seed money as did Mike, and hire programmers as did Diane. Neither Mike nor Diane did precisely that, but, using a bit of imagination, Claire selects cases from previous stories and recombines them to generate a new plan. (See Figure 2.)

In our model, plans are paths in a directed graph. The vertices in the graph represent *positions*, which are vectors of real numbers. A *case* is an edge in the graph, leading from one position to another. For example, in Example 1 we may think of binary variables $x^j \in \{0, 1\}$, each indicating whether the traveler is in a given location j. Suppose that Paris corresponds to $j = 1$, Bangkok and Hanoi – to $j = 2$ and $j = 3$, respectively, whereas Melbourne and San Francisco – to $j = 4$ and $j = 5$. Thus, Anne starts with the position

[1] This idea is the basis of the model of "case-based planning" in Gilboa and Schmeidler (2001). We explain the relationship between the present model and the previous one in the Discussion section.

$x = (1, 0, 0, 0, 0)$ and she wishes to reach another position, $y = (0, 0, 1, 0, 0)$. Jean's story is given by the sequence

$$(1, 0, 0, 0, 0) \to (0, 1, 0, 0, 0)$$
$$(0, 1, 0, 0, 0) \to (0, 0, 0, 1, 0)$$

indicating that it is possible to reach $(0, 1, 0, 0, 0)$ from $(1, 0, 0, 0, 0)$. Patricia's story is given by

$$(0, 0, 0, 0, 1) \to (0, 1, 0, 0, 0)$$
$$(0, 1, 0, 0, 0) \to (0, 0, 1, 0, 0)$$

suggesting that one can reach $(0, 0, 1, 0, 0)$ from $(0, 1, 0, 0, 0)$. Anne takes these two cases and devises her plan

$$(1, 0, 0, 0, 0) \to (0, 1, 0, 0, 0)$$
$$(0, 1, 0, 0, 0) \to (0, 0, 1, 0, 0)$$

where each edge has appeared in a previous story, though the entire path has not. (See Figure 3.)

Let us now turn to Example 2. It has a very similar structure, but it involves non-binary variables denoting amounts of money. Specifically, in order to model this example by transitions between positions, one may use variables denoting "available funds" and "debt", to be able to model the act of borrowing money. Let us focus on three variables: x^1, x^2, x^3, where x^1 denotes available funds, x^2 – debt, and x^3 – the number of programmer teams on the payroll. A position will have additional variables describing the economic status and activities of the various agents, but, for the sake of simplicity, we leave their values unspecified. Claire starts out with a position $(0, 0, 0, ...)$. She knows that Mike had the following case as part of his story:

$$(0, 0, 1, ...) \to (1, 1, 1, ...)$$

that is, he took a loan of \$1M. The unspecified variables are assumed to be unchanged by the transition from one position to another. In fact, Claire may not know their values. However, she assumes that, whatever these values were, they did not change when Mike took a loan. Next, Claire knows of Diane's firm hiring programmers, which can be modeled by

$$(10, 0, 0, ...) \rightarrow (9, 0, 1, ...)$$

Claire comes up with the plan

$$(0, 0, 0, ...) \rightarrow (1, 1, 0, ...)$$
$$(1, 1, 0, ...) \rightarrow (0, 1, 1, ...)$$

allowing her to raise funds, as did Mike, and hire programmers, as did Diane.

1.2 Similarity of Cases

One important difference between the two examples is the following. In Example 1, each case used in the new plan has already appeared in a previous story. By contrast, this does not hold in Example 2: for example, Claire plans to reach $(0, 1, 1, ...)$ from $(1, 1, 0, ...)$ based on Diane's experience, who reached $(10, 0, 1, ...)$ from $(9, 0, 0, ...)$. The latter isn't the same case Claire plans to use. However, Claire might argue that the cases are *similar*. Specifically, they are similar in the sense that the differences between the target and the source positions are the same:

$$\begin{aligned} (0, 1, 1, ...) - (1, 1, 0, ...) &= (-1, 0, 1, 0, ..., 0) \\ (10, 0, 1, ...) - (9, 0, 0, ...) &= (-1, 0, 1, 0, ..., 0) \end{aligned}$$

In our model, case (x, y) is *similar* to case (z, w) if $y - x = w - z$.

Defining similarity by equality of differences is surely restrictive. One may well generalize the model by introducing an abstract similarity relation S between cases, so that $((x, y), (z, w)) \in S$ would indicate that knowledge of case (z, w) may make the planner think of case (x, y). However, differences between real-valued vectors seem to be a simple and intuitive model of similarity with the following features:

(i) The differences ignore variables that are unchanged between two positions. For instance, in Example 2, hiring programmers is captured by the difference vector $(-1, 0, 1, 0, ..., 0)$ in which the level of debt and all other variables are unchanged between the two positions. Clearly, there is an implicit assumption in considering such cases as similar: basically, it is assumed that, if the source and the target position are equal, it does not matter what they are equal to. This may not be the case. For example, it is possible that a variable denoting time is equal in the source and target positions of a past case, as well as in a future case, but that experience from the past is not very relevant to the future. However, we attempt to capture the imaginative, creative process of coming up with new plans, and it seems plausible that, for this process, variables whose values do not change between two positions would be "canceled out". More careful development of a detailed plan would need to verify that the apparent similarity indeed offers a sound enough basis for action.

(ii) Difference vectors add up along a path in a graph in an intuitive way. In Example 1, for instance, Jean's story suggested that

$$(0, 1, 0, 0, 0) - (1, 0, 0, 0, 0) = (-1, 1, 0, 0, 0)$$

was possible, that is, that one could reach Bangkok from Paris. Patricia's story suggested that

$$(0, 0, 1, 0, 0) - (0, 1, 0, 0, 0) = (0, -1, 1, 0, 0)$$

was also possible, that is, that one could reach Hanoi from Bangkok. The concatenation of these cases into a path is algebraically represented by addition of difference vectors:

$$(-1, 1, 0, 0, 0) + (0, -1, 1, 0, 0) = (-1, 0, 1, 0, 0)$$

capturing the possibility of a path from Paris to Hanoi. More generally, since we can write

$$(z - x) = (z - y) + (y - x),$$

thinking of the difference between two positions, $(z - x)$, as a "possible transition" captures the notion that there is a path between x and z, whether by a single case or by concatenation of cases.

(iii) Finally, differences are quite natural when one considers economic resources such as money, labor hours, and the like. In Example 2, having observed a case in which the position $(9, 0, 1, ...)$ was reached from $(10, 0, 0, ...)$, we can infer that \$1M can suffice to hire a team of programmers. Thus, if one starts out with a different amount of money, K, some debt level D, and p teams of programmers, one could plan to subtract 1 from K and augment p by 1, that is, to have a case starting at $(K, D, p, ...)$ and leading to $(K - 1, D, p + 1, ...)$.[2]

Having said that, the definition of similarity by equality of differences is certainly no more than a simple model. In particular, it should be generalized to differences that are close in some sense (but not necessarily identical), and to notions of similarity that depend on the entire vectors under discussion.[3]

A plan using only cases that have been encountered in the past in their entirety will be referred to as "simple". Thus, Example 1 involves a simple plan, whereas Example 2 does not. The definition of a "simple plan" depends on the level of detail at which positions are described. Indeed, one can choose a level of detail that would make each position unique, thereby excluding the possibility of simple plans (for the future). For example, in Example 1 we modeled only location variables, and did not include in the description of a position details such as the identity of the protagonist, the exact time, and so forth. A more detailed description of a position could be "Jean was in Paris in 2023, and then Jean was in Bangkok in 2023". If Anne wishes to use this case in her planning, she would have to replace "Jean" by "Anne" and "2023" by "2025". The cases involved will no longer be identical, but they will be similar (because their differences remain identical).

1.3 Models of Reasoning

Perfect Bayesian rationality, as in classical dynamic programming (Bellman, 1954, 1957), assumes that the decision maker can fully conceive of all future

[2]The difference between two positions generates a vector in which negative entries can be viewed as inputs, and positive ones – as outputs, in a matter that brings to mind the definition of a "technology" in a general equilibrium model. (See, for example, Mas-Colell-Whinston-Green, 1995).

[3]See the Discussion section below.

eventualities, as well as all possible actions she may take at any step, and she has Bayesian beliefs about the evolution of her "position" as a result of her actions. Pushed to an extreme, perfect Bayesian rationality assumes that the decision maker (behaves as if she) had conceived of all eventualities before she was born, as it were, and all the reasoning that is left for her to do is to update probabilities using Bayes's rule.[4] This model is cognitively very demanding, and it may not shed much light on the process of planning. Specifically, it is hard to capture the notion of "imagination" in such a model.

By contrast, inspired by the notion of case-based reasoning (Schank, 1986, Kolodner, 1992), case-based decision theory (Gilboa and Schmeidler, 1995) was offered as a formal model of decision making under uncertainty, in which the decision maker is assumed to be driven by successes and failures of the available acts in similar past cases. In this model, the decision maker makes no explicit attempt to predict the future. As a result, the model can hardly capture innovation or imaginative thinking: the decision maker is doomed to repeat previous choices.

The present model may be viewed as offering a middle ground between these extremes: it allows the reasoner to come up with new plans, but the building blocks of these plans are restricted to those encountered in known cases. Thus, the reasoner is not assumed to conjure up cases that have nothing to do with those she has observed, but she is supposed to have considerable freedom in putting past cases together in new ways.[5]

1.4 Imaginative Plans vs. Detailed Strategies

Our model may be viewed as generating "proto-plans" rather than concrete, detailed, strategies. There are at least four ways in which the "plans" in our model fall short of concrete strategies:

1. **Agency**: The model we suggest does not have an explicit ingredient of

[4] Savage (1954) commented on this point. Despite his somewhat humorous tone, this is the common assumption in models of decision and game theory. (See, for instance, Fudenberg and Tirole, 1991, or Myerson, 1991).

[5] This is in line with Schank's (1986) of creativity, putting an emphasis on case-based reasoning as the basis of presumably original thought.

an act. Moving from one position to another is done without specifying who acts and what they do in order to bring about the change (from one position to another).

2. **Utility**: The model does not mention how well off the planner is in various positions. The focus on reaching a certain position y implicitly suggests that y is a desirable position to be at, but there is no quantification of how desirable it is, and how desirable or undesirable other positions are – including those along the way to y as well as other positions one may find oneself at.

3. **Uncertainty**: Another important aspect that the model is silent on is the uncertainty about the unfolding of a plan. How certain is the planner that one position would follow another? What would happen if other positions materialize, rather than the intended one? It is natural to augment the model with additional paths, starting at different positions, to capture the idea of "Plan B", that is, of fallback options in case the plan is not executed as imagined. Yet, the model does not offer any probabilistic assessments of the likelihood of different scenarios.

4. **Time**: Finally, the model is silent also on the amount of time one expects to spend in various positions along the plan's path, as well as on the time a final position will be enjoyed before something else happens.

Each of the above could be a reason for a plan to fail. For example, the planner might believe that she will be able to determine the values of variables that will turn out to be under someone else's control. She may find that her plan was rather optimistic, and that she failed to take into account random shocks. Likewise, she may have to spend a long time in undesirable positions before moving on to the more desirable ones.

Clearly, a rational model of developing a strategy should not ignore these factors.[6] Our goal here is not to suggest a new decision theory, replacing notions of players, acts, outcomes, utilities, or probabilities. Our goal is to model the initial stages of planning. A plan as modeled here would not suffice for

[6] Indeed, automated planning attempts to address these issues. (See, for instance, Levitt, Kunz, and Kartam, 1987, Malik, Nau, and Traverso, 2004, and Cimatti, Pistore, and Traverso, 2008.)

a rational agent to start acting; rather, the agent would be expected to complement all the missing details in order to come up with a workable strategy. However, these details might cause distraction and hamper imaginative thinking. In this note, we seek to model the creative process in which the planner allows her imagination to roam about without dotting her i's and crossing her t's.[7]

1.5 Outline

The next section presents the (very simple) model, and a complexity result. It states that, given a memory of past cases, it is computationally easy to determine whether these cases suggest a simple plan to reach a position y from a given position x, but that it is computationally hard to solve this problem for general (not necessarily simple) plans. We view this result as an indication that it is hard to come up with plans: simple plans are rather limited, and, as mentioned above, a plan can be simple only if some variables are omitted from the model. Thus, the fact that it is difficult to find a plan algorithmically suggests an explanation for the fact that imagination requires a special knack: it involves dealing with a problem for which no efficient algorithm is known. In Section 3 we further discuss the meaning of this result and put it in the context of learning and expertise in case-based models.

2 Model

A *position* is a vector $x = (x^1, ...x^m) \in X \equiv \mathbb{R}^m_+$. It is assumed that all variable values are nonnegative, which suits many economic contexts. However, this assumption can be relaxed to permit values that are simply bounded from below. A *case* is a pair $c \in C \equiv X \times X$ where $c = (x, y)$ describes an actual

[7] It has been noted (Boorman, 2004, 2024 [see footnote 1 on p. 47] and Geanakoplos, 2004) that game theory does not offer a distinction between "tactics" and "strategy". In a sense, the everyday notion of a "strategy" may be closer to the "plans" in our model: both have to be supplemented by details of implementation, and evaluated for their feasibility and cost. It should be noted, however, that the formal notion of a "strategy" in game theory refers to the detailed plan of action, rather than to the bold-strokes proto-plan that is often indicated by the term in everyday parlance.

(past) or imagined (future) transition from one position, x, to a consecutive one, y. Two cases $(x, y), (z, w)$ are *similar* if

$$y - x = w - z$$

We are given a finite set of cases $M \subset C$, interpreted as the memory of cases that have been observed in the past.

A *M-plan to reach y from x* is a path $((x_0, x_1), (x_1, x_2), ..., (x_{k-1}, x_k))$ with $x_0 = x$ and $x_k = y$ such that, for every $1 \leq i \leq k$, there exists $c_i \in M$ for which (x_{i-1}, x_i) and c_i are similar. The *length* of such a plan is k. An M-plan is *simple* if the above holds with (x_{i-1}, x_i) being identical to c_i for each i.

Problem 1 SIMPLE-PLAN: Given memory M, $x, y \in X$, and $k \geq 1$, does there exist a simple M-plan to reach y from x of length not exceeding k?

Problem 2 PLAN: Given memory M, $x, y \in X$, and $k \geq 1$, does there exist an M-plan to reach y from x of length not exceeding k?

It is natural to pose the corresponding problems without reference to the length of the plan.[8] That is, to consider the following problems:

Problem 3 SIMPLE-PLAN*: Given memory M and $x, y \in X$, does there exist a simple M-plan to reach y from x?

Problem 4 PLAN*: Given memory M and $x, y \in X$, does there exist an M-plan to reach y from x?

We now state

Theorem 1 *Problems SIMPLE-PLAN and SIMPLE-PLAN* are polynomial, but Problem PLAN is NP-Complete and PLAN* is NP-Hard.*

We interpret the standard notions of "Polynomial" vs. "NP-Hard" problems as "easy" vs. "difficult". This interpretation may well be questioned.

[8] We thank Mateus Hiro Nagata for raising this question.

First, these notions are defined asymptotically. A problem whose computational complexity explodes in an exponential fashion can still be manageable for small inputs. On the other hand, a problem whose complexity is bounded by a polynomial may be unwieldy if the input size is large, and even more so if the degree of the polynomial turns out to be high. Second, the definitions involve worst-case complexity for a perfect solution, while in reality we might be interested in average-case complexity and/or in approximated solutions. Third, a problem might be polynomial but finding a polynomial algorithm that solves it may be a difficult mathematical problem.

In our model these computational complexity classes are used as metaphors. Our focus isn't the development of algorithms that can solve the planning problems in practice. Indeed, the statement of the problems in our model is too idealized for such a goal to begin with. Rather, we view the model as capturing some features of human reasoning.[9] For that purpose, it appears that a polynomial problem can be viewed as simple to solve, especially if the degree of the polynomial isn't high and the algorithm itself is intuitive. By contrast, an NP-Hard problem tends to be such that humans facing it feel at a loss.

There is another important feature, specific to NP-Complete problems, which seems to resonate with our thinking about human problem solving, and it is the distinction between looking for a solution and verifying that a proposed solution is indeed valid. NP problems are such that the latter – verification of a proposed solution – is an easy task. However, NP problems that are also NP-Complete are considered to be difficult to solve when one does not get a hint, as it were, for a solution. This appears to be a reasonable model of an "A-ha" moment that people might experience when someone suggests a solution to them, and, post hoc, it seems obviously correct.

[9] To an extent, the analysis might be relevant to deep learning systems, too.

3 Discussion

3.1 Information, Insight, and Imagination

Case-based models of prediction and decision suggest a natural distinction among three types of knowledge/skill/expertise:

1. **Information**: In models that are based on knowledge of past cases, there is a clear sense in which one may "know more": one may have a larger database/memory of cases (in the sense of set inclusion). Clearly, this is a partial ordering over databases. Yet, it gives a clear idea about the benefits of experience. From the viewpoint of "hard" information, learning can take the form of accumulating more cases, and an expert can be expected to know more cases.

2. **Insight**: Given a database of cases, the reasoner can use them in various ways, and, in particular, may employ different similarity functions to find the relevant past cases for a problem at hand. Gilboa, Lieberman, and Schmeidler (2006) and Gayer, Gilboa, and Lieberman (2007) suggested the process of "second-order induction", by which one learns the appropriate similarity function from the data.[10] Argenziano and Gilboa (2019) showed that finding the empirically-optimal similarity function is an NP-Complete problem. Coming up with such a function can be referred to as *insight*: it amounts to suggesting a way to look at the database which "suddenly" makes sense and exposes its structure. Insights are often easy to verify once proposed, but hard to find without guidance – as are solutions to NP-Complete problems.[11]

Insight is a type of expertise that differs from information: in asking an expert for the "right" way to evaluate similarity between cases, one does not ask for hard, verifiable information. Rather, one asks for the kind of knowledge that, in hindsight, one thinks one could have figured out on one's own based

[10] It is considered to be "second-order" because relying on past cases is already a form of induction, and learning the similarity function is, in a way, learning how to learn.

[11] While it is easy to verify that a certain similarity function organizes the database "neatly", it is generally hard to verify that there is no other function that does an even better job. This, again, is similar to NP-Complete problems that are yes/no ("decision") rather than optimization problems.

on available data. Yet, the fact that there are no efficient algorithms that are guaranteed to find the appropriate similarity function suggests that coming up with new insights is a skill that is not well-understood.

3. **Imagination**: Finally, the third type of knowledge/expertise is the ability to imagine new plans based on past cases, as modeled here. In this model imagination is restricted to selecting and combining cases that are similar to existing ones. When restricting attention to simple plans, that is, when past cases are used in their entirety, we noted that the problem of finding a plan is computationally easy. This suggests that the process of coming up with simple plans may be well-understood and that this type of imagination isn't too mysterious. But when we upgrade the model and allow past cases to bring to mind *similar* (rather than identical) future ones, the problem becomes NP-Complete. This means that it will generally be difficult to come up with a plan, while it will be easy to verify that a suggested plan makes sense.[12]

Imagination is therefore a third type of expertise, differing from information and from insight: it's not about knowing past cases, nor about knowing how to organize them – but about knowing how to use them for future plans. The NP-Completeness result suggests that imagination, or creative thinking, is hard to teach and to learn.

To sum, of the three types of thinking, the first is somewhat technical: it requires gathering information and preserving it. The other two require some talent or expertise that we don't seem to understand very well.

3.2 Brainstorming

If the act of planning is conceived of as looking for paths in a graph, one can model brainstorming by introducing several reasoners/planners, each having her own set of cases in memory, working together to figure out a plan. We may think of each reasoner i, equipped with memory M_i, coming up with various partial paths in the positions graph. For instance, if the team is looking for a

[12] Again, it will also be generally hard to determine if there is no other, better plan. Specifically, a planner may come up with a plan of length k to reach y from x, but there may be a shorter plan that accomplishes this task.

plan to reach y from x, and a reasoner can think of a plan to reach w from z, she may contribute this idea even if $w \neq y$ and $z \neq x$. Such a partial plan can be viewed as saying, "If one of you can figure out how to reach z from x, and someone else has an idea for reaching y from w, here's a way to complement these into a plan".

Osborn (1953) introduced the notion of brainstorming, emphasizing the advantage of collaboration and judgment deferral, among other principles. In our model, the deferral of judgment can be related to the absence of time and uncertainty considerations, seeking a general plan without the critical questions of "how long will it take us?" and "... and what if not?". Clearly, our model can explain the advantage of larger teams, who will be aware of a larger set of cases.[13]

Can a brainstorming team be too large? The answer may be in the affirmative for various reasons.[14] One of these is that different people may be aware of different sets of variables. When people brainstorm, each will become aware of all the variables brought up by everyone else, and the similarity between cases might decrease. In our simplified model, for a case (x, y) to be similar to (z, w), we have to have $y - x = w - z$. If we add a variable x^{m+1} to the description of (all) positions, this equality may no longer hold. Hence, adding a variable can only shrink the set of pairs of cases that are similar to each other and, consequently, when many people are involved in a team, it might be difficult to see similarity between cases.

Our model can therefore be used to analyze optimal group size for planning. Clearly, in order to obtain any results, many assumptions need to be imposed on the probability that each planner can think of each past case and of each variable, on the correlations among different planners, on the length of the plan, and so forth.

[13]See also McGuire (1997) who explores a variety of heuristics for promoting creative thinking.

[14]See Brooks (1995).

3.3 Heuristics

Finding a plan to transition from one position to another can be computationally challenging, particularly when imagination and creativity are required. In such instances, heuristic approaches may be useful. Below, we present two methods that simplify the planning task by focusing on practical partial solutions. Both methods reduce the problem's dimensionality, making it easier to determine whether a solution exists.

Lexicographic Method

The lexicographic method prioritizes variables based on a predefined order of importance. The planner first focuses on the highest-priority variable, attempting to transition it to its target value. If no solution can be found at this stage, the process terminates. Otherwise, once the highest-priority variable reaches its goal, the planner advances to the next variable while ensuring the previously solved higher-priority variables remain unchanged. If it fails at any step, planning ceases. A key outcome of this approach is that, although it might not achieve target values for all variables, it can successfully match a subset of them (when such solutions exist). When the variables are causally independent, it is more likely for there to be cases in memory that do not alter the values of higher-priority variables when adjusting lower-priority ones. As a result, the method can reach more of the overall objective before terminating. The lexicographic method therefore works best when variables can be naturally prioritized and are mostly independent.

Partitioned Aggregation Method

The partitioned aggregation method addresses complexity by grouping variables into sub-partitions and aggregating their values within those groups. Both the initial position and the target position are then represented by these aggregated values. The planning objective is to find a path from the aggregated initial state to the aggregated target state. Since the number of variables is reduced, the resulting search is simpler. If a path is found for a given partition scheme, the search can be refined by using a finer partition. Conversely, if no solution is found, one can revert to a coarser partition, where it is easier to verify if a solution exists. This method better fits situations where there is a natural way to group variables. For example, if locations are represented by indicator vectors (as in Example 1), it makes sense to partition variables by region, where summing within an element of the partition will represent presence in a broader region (e.g., Far East) rather than in a specific location. Similarly, in Example 2, different funding resources can be spelled out individually, or be combined into a single variable representing total debt, simplifying the planning process while retaining meaningful information. However, if the variables in question are all distinct and unrelated, partitioned aggregation becomes less useful.

3.4 Related Literature

The model presented here is inspired by the notion of "case-based planning" in Gilboa and Schmeidler (2001, Ch. 5). There are two main differences between the models. On the one hand, the present model has more structure than the previous one: whereas in Gilboa and Schmeidler (2001) a "position" is an abstract entity, in the current model it is a vector of nonnegative numbers. In particular, this specification allows us to define similarity of cases by equality of differences. On the other hand, the present model is completely silent on action, probability, time horizon, and even agency. We believe that abstracting away from these details offers a better model of imaginative thinking, a cognitive step which precedes the development of a concrete strategy.

3.5 Extensions

In our model, similarity of positions is defined solely by their differences. Thus, if two positions are equal on a subset of coordinates, it does not matter what they are equal to. This is a sort of "independence" assumption, and, as mentioned above, it may not be very compelling. For example, if one variable denotes time, measured in years, a case that happened in 1066 may have little relevance to the same case happening in 2025. This can have an effect on two mental processes: first, the judgment of the probability that a position would evolve into another one along a given case. Second, the likelihood that a past case would pop up in a planner's memory. The former need not bother us here, as we are dealing with a model of imaginative thinking, presumably a prerequisite to careful strategizing. However, the latter process is relevant also to the model of free associative planning: a case that happened a millennium ago may not occur to the planner in her thinking about a problem at present.

It therefore seems reasonable to extend the notion of similarity from equality of differences to an abstract function defined on pairs of cases. This would allow similarity judgment to rule two cases as dissimilar even if they connect positions with identical differences, and also to allow some degree of similarity even if the differences are not identical.

Extending the model to include a more general similarity relations, and perhaps also continuous similarity functions, raises some modeling questions. In particular, which pairs of cases are brought to the planner's mind? Does any positive similarity value suffice for the planner to conceive of a possible case? Is there a similarity threshold that is needed for a case to be imagined, given another case? Can the similarity function that is used for probability assessments differ from the one used for cases popping up in the planner's imagination? We leave these questions for future research.

Appendix: Proof of Theorem

3.6 SIMPLE-PLAN is Polynomial

Let there be given a memory M, two positions $x, y \in X$, and $k \geq 1$. Denote the members of M by $c_i = (z_i, w_i)$ for $i = 1, ..., n$ (with $n = |M|$). Consider the graph whose nodes are $V \equiv \{z_i\}_{i \leq n} \cup \{w_i\}_{i \leq n}$ and whose set of edges is M. If x or y are not in V, there can be no M-plan to reach y from x. If they both are, we only have to check whether there is a path of length no larger than k in V, from x to y – which can obviously be done in Polynomial time.

3.7 SIMPLE-PLAN* is Polynomial

Let there be given a memory M and two positions $x, y \in X$. Let $n = |M|$. An M-plan from x to y exists iff such a plan exists of length that doesn't exceed $k = 2n$. (This is true because, in any directed graph, a path between two nodes exists iff there exists a path between them with no cycles, and the length of such a path cannot exceed the number of nodes in the graph.) We therefore proceed as in the algorithm for SIMPLE-PLAN with the same input and $k = 2n$.

3.8 PLAN is NP-Complete

We now turn to show that PLAN in NP-Complete. It is straightforward that PLAN is in NP: given a memory M and a sequence $((x_0, x_1), (x_1, x_2), ..., (x_{k-1}, x_k))$ with $x_0 = x$ and $x_k = y$, verifying that the latter is an M-plan takes $O(nk)$ computation steps. (We assume that algebraic operations on vectors of length m take 1 time unit.)

To show that PLAN is NP-Complete, we reduce the Hamiltonian Cycle problem (in a directed graph) to PLAN. Let there be given a directed graph $G = (V, E)$ with vertices $(v_1, ..., v_r)$ and edges $E \subset V^2$. We assume, w.l.o.g., that $(v_i, v_j) \in E$ implies $i \neq j$.

We construct the following instance of PLAN. We set $m = r$, so that each coordinate corresponds to a vertex in G and choose memory size to be $n = |E|$.

Memory M is defined by

$$M = \left\{ \left(e^i, 2e^j\right) \mid (v_i, v_j) \in E \right\}$$

where e^i is the i-the unit vector in $\mathbb{R}^m$. Let

$$\begin{aligned} x &= e^1 = (1, 0, ..., 0) \\ y &= e^1 + \sum_{i \leq m} e^i = (2, 1, ..., 1) \end{aligned}$$

Finally, let $k = r$. Observe that this construction can be performed in polynomial time.

We claim that $G = (V, E)$ has a Hamiltonian cycle iff there exists a plan to get from x to y of length k.

Only if: Assume that a Hamiltonian cycle exists. Thus, there is a permutation $\pi : \{1, ..., r\} \to \{1, ..., r\}$ with $\pi(1) = 1$ such that, for every $i \leq r$, $\left(v_{\pi(i)}, v_{\pi((i+1) \bmod r)}\right) \in E$ (so that, for $i < r$, $\left(v_{\pi(i)}, v_{\pi(i+1)}\right)$ is an edge in E, and so is $\left(v_{\pi(r)}, v_1\right)$). Consider the sequence $((x_0, x_1), (x_1, x_2), ..., (x_{k-1}, x_k))$ defined by

(i) $x_0 = x \, (= e^1)$

(ii) for $1 \leq i \leq k (= r)$, $x_i = x_{i-1} + 2e^{\pi((i+1) \bmod r)} - e^{\pi(i)}$.

This sequence is an M-plan, because, for each $1 \leq i \leq k$, the case (x_{i-1}, x_i) is similar to a case in M:

$$x_i - x_{i-1} = 2e^{\pi((i+1) \bmod r)} - e^{\pi(i)} = 2e^t - e^s$$

for some $s, t \leq r$ such that $(v_s, v_t) \in E$. Finally, observe that, because each vertex in V appears exactly once in the cycle,

$$x_k = x_0 + \sum_{1 \leq i \leq k} 2e^{\pi((i+1) \bmod r)} - e^{\pi(i)} = x_0 + \sum_{1 \leq i \leq k} e^i = y$$

so that this M-plan reaches y from x.

If: Next assume that an M-plan $((x_0, x_1), (x_1, x_2), ..., (x_{q-1}, x_q))$ exists, with $x_0 = x$ and $x_q = y$, and $q \leq k$.[15] For each $1 \leq l \leq q$, there exists a

[15] As will be shown below, if there is a plan with $x_0 = x$ and $x_q = y$, then $q = k$ follows from a simple summation argument, without relying on $q \leq k$. However, this observation doesn't greatly simplify the current proof.

case $c_l = (z_l, w_l) \in M$ such that $x_l - x_{l-1} = w_l - z_l$. Since the cases in M are defined by edges in G, for each $1 \leq l \leq q$ there are $s(l), t(l) \leq r$ such that $\left(v_{s(l)}, v_{t(l)}\right) \in E$ and

$$x_l - x_{l-1} = w_l - z_l = 2e^{t(l)} - e^{s(l)}$$

Let us refer to the list $(s(l))_{l \leq q}$ as the sources and to $(t(l))_{l \leq q}$ as targets.

Claim 1: Each index $1 \leq j \leq k$ appears exactly once in $(s(l))_{l \leq q}$ and exactly once in $(t(l))_{l \leq q}$; in particular, $q = k$.

Proof: We start with the list of targets. If, for a given vertex $v_j \in V$, v_j does not appear in the list of targets, that is, if $v_j \neq v_{(t(l))}$ for all $l \leq q$, we have to have $y^j \leq x^j$: the only way in which the j-th coordinate in the position can increase (between x and y) is when j appears in the target list. But this contradicts the definition of x, y, by which $y^j = x^j + 1$. Hence each j appears in $(t(l))_{l \leq q}$ at least once. Given that $q \leq k$, we also get $q = k$, and, further, that each j appears in $(t(l))_{l \leq k}$ exactly once.

Next, consider the list of sources $(s(l))_{l \leq k}$. Note that, whenever a node v_j appears as the source of an edge, the j-th coordinate of the position is decreased by 1. Since it increases only once by 2, and overall it has to increase by 1, it has to be decreased by 1 at most once – and, indeed, precisely once. $\square$

Claim 2: There exists $2 \leq q \leq k$ such that $\left(\left(v_{s(l)}, v_{t(l)}\right)\right)_{l \leq q}$ is a simple cycle in G,[16] starting and ending at v_1. That is, (i) $s(1) = t(q) = 1$; (ii) for all $1 \leq l < q$, we have $s(l+1) = t(l)$; and (iii) $(s(l))_{l \leq q}$ are distinct.

Proof: We start with $s(1)$. We have $x_1 = x_0 + 2e^{t(1)} - e^{s(1)}$ (with $t(1) \neq s(1)$). This means that, for x_1 to be nonnegative, we have to have $x_0^{s(1)} \geq 1$ and this holds only if $s(1) = 1$.

Next, consider $s(2)$. We have $x_2 = x_1 + 2e^{t(2)} - e^{s(2)}$ (again, with $t(2) \neq s(2)$). Thus, the $s(2)$ coordinate of the position decreases by 1 as we move from x_1 to x_2, and for x_2 to be nonnegative we have to have $x_1^{s(2)} \geq 1$. Given the $s(2) \neq s(1)$ (as all $(s(l))_{l \leq k}$ are distinct by Claim 1), it has to be the case

[16] When referring to a "simple cycle" or a "simple path" we use the standard graph-theoretic definition, namely a cycle or a path that does not cross itself. This standard terminology is unrelated to our notion of a "simple plan".

that $s(2) = t(1)$. We continue by induction (on $p \geq 1$) to show that, as long as $t(l) \neq 1$ for $l \leq p$, $\left(\left(v_{s(l)}, v_{t(l)}\right)\right)_{l \leq p}$ is a simple path in G (starting at v_1). Assuming that this is the case for $\left(\left(v_{s(l)}, v_{t(l)}\right)\right)_{l \leq p}$, we consider $s(p+1)$. As above, $s(p+1) \neq s(1)$ because all $(s(l))_{l \leq k}$ are distinct by Claim 1. Since $x_{p+1} = x_p + 2e^{t(p+1)} - e^{s(p+1)}$ (and $t(p+1) \neq s(p+1)$), nonnegativity of x_{p+1} implies that $x_p^{s(p+1)} \geq 1$. Since $s(p+1) \neq s(1)$, the only way in which $x_p^{s(p+1)}$ can be strictly positive is that $s(p+1) = t(l)$ for some $l \leq p$. However, for $l \leq p-1$ we already know, by the induction assumption, that $t(l) = s(l+1)$ and since $(s(l))_{l \leq k}$ are all distinct, we cannot have $t(l) = s(p+1)$ as well. We conclude that $s(p+1) = t(p)$.

Next, we consider $t(p+1)$. If it so happens that $t(p+1) = 1$, we found a cycle of length $p+1$ from 1 to itself, and it has to be simple as all the targets are distinct. If not, we proved the induction claim for $p+1$. It only remains to note that after at most k steps we run out of new vertices and the path has to become a cycle. □

Claim 3: The cycle found in Claim 2 contains all vertices, that is, $q = k$.

Proof: Assume not. Then we have a cycle of length $q < k$: $\left(v_1, v_{s(2)}, ..., v_{s(q)}, v_1\right)$ so that $\left(v_1, v_{s(2)}, ..., v_{s(q)}\right) = \left(v_{s(l)}\right)_{l \leq q}$ and $\left(v_{s(2)}, ..., v_{s(q)}, v_1\right) = \left(v_{t(l)}\right)_{l \leq q}$. We then have

$$x_q = x_0 + \sum_{1 \leq l \leq q} 2e^{t(l)} - \sum_{1 \leq l \leq q} e^{s(l)} = x_0 + \sum_{1 \leq l \leq q} e^{s(l)}$$

and, for every j such that $j \notin (s(l))_{l \leq q}$ we have $x_q^j = 0$. Consider $x_{q+1} = x_q + 2e^{t(q+1)} - e^{s(q+1)}$. Because all $(s(l))_{l \leq k}$ are distinct, $s(q+1)$ isn't among $(s(l))_{l \leq q}$, and thus $x_q^{s(q+1)} = 0$ and $x_{q+1}^{s(q+1)} < 0$ – a contradiction. □

We have therefore shown that if there exists an M-plan reaching y from x, the graph G has a Hamiltonian cycle. This concludes the proof of the theorem.

3.9 PLAN* is NP-Hard

PLAN* is not in NP, because the size of a potential answer is not bounded (and, in particular, not bounded by any polynomial of the input).

To see that the problem is NP-Hard, we use the same reduction as in the case of PLAN. The main point to observe is that for any case in $M =$

$\{ (e^i, 2e^j) \mid (v_i, v_j) \in E \}$ the sum of all coordinates in $2e^j - e^i$ is 1. That is, denoting $\mathbf{1} = (1, 1, ..., 1)$,

$$\mathbf{1}{\cdot}2e^j - \mathbf{1} \cdot e^i = 1$$

and thus, for any plan $((x_0, x_1), (x_1, x_2), ..., (x_{k-1}, x_k))$ we have, for every $1 \leq l \leq k$, $\mathbf{1}{\cdot}x_{l+1} - \mathbf{1} \cdot x_l = 1$, and therefore $\mathbf{1}{\cdot}x_k - \mathbf{1} \cdot x_0 = k$. Because

$$\mathbf{1}{\cdot}y - \mathbf{1} \cdot x = m$$

we conclude that an M-plan reaching y from x exists iff such a plan exists of length $k = m$. We proceed as in the proof above to argue that such a plan exists iff the original graph has a Hamiltonian path. □

4 Bibliography

Argenziano, Rossella and Itzhak Gilboa (2019), "Second-Order Induction in Prediction Problems", *Proceedings of the National Academy of Sciences*, **116**: 10323-10328. https://doi.org/10.1073/pnas.1901597116.

Bellman, Richard (1954), "The theory of dynamic programming", *Bulletin of the American Mathematical Society*, **60**: 503-516.

Bellman, Richard (1957), *Dynamic Programming.* Princeton University Press.

Boorman, Scott (2004), personal communication.

Boorman, Scott (2024), *Three Faces of Sun Tzu Analyzing Sun Tzu's Art of War, A Manual on Strategy*, Cambridge University Press.

Brooks, Frederick (1995), *The Mythical Man-Month: Essays on Software Engineering.* Addison-Wesley.

Cimatti,Alessandro, Marco Pistore, and Paolo Traverso (2008), "Automated Planning", *Foundations of Artificial Intelligence*, **3**: 841-867.

Fudenberg, Drew and Jean Tirole (1991), Game Theory. Harvard University Press.

Geanakoplos, John (2004), personal communication.

Gilboa, Itzhak, Offer Lieberman, and David Schmeidler (2006), "Empirical Similarity", *Review of Economics and Statistics*, **88**: 433-444.

Gayer, Gabrielle, Itzhak Gilboa, and Offer Lieberman (2007), "Rule-Based and Case-Based Reasoning in Housing Prices", *BE Journals in Theoretical Economics*, **7**.

Gilboa, Itzhak and David Schmeidler (1995), "Case-Based Decision Theory", *The Quarterly Journal of Economics*, **110**: 605-639.

Gilboa, Itzhak and David Schmeidler (2001), *A Theory of Case-Based Decisions.* Cambridge University Press.

Kolodner, Janet L. (1992), "An Introduction to Case-Based Reasoning", *Artificial Intelligence Review,* **6**: 3-34.

Levitt, Raymond E., John C. Kunz, and Nabil A. Kartam (1987), "Using Artificial Intelligence Techniques for Automated Planning and Scheduling", *Proceedings of the 4th ISARC*, Haifa, Israel: 27-52.

Malik, Ghallab, Dana S Nau, and Paolo Traverso (2004), *Automated Planning: Theory and Practice.* Morgan Kaufmann.

McGuire, William (1997), "Creative Hypothesis Generating in Psychology: Some Useful Heuristics", *Annual Review of Psychology*, **48**: 1-30.

Myerson, Roger (1991), Game Theory: Analysis of Conflict. MIT Press.

Osborn, Alex F. (1953), *Applied Imagination: Principles and Procedures of Creative Problem Solving.* Charles Scribner's Sons.

Mas-Colell, Andreu, Michael D. Whinston, and Jerry R. Green (1995), *Microeconomic Theory.* Oxford University Press.

Savage, Leonard J. (1954), *The Foundations of Statistics.* New York: John Wiley and Sons. (Second addition in 1972, Dover)

Schank, Roger C. (1986), *Explanation Patterns: Understanding Mechanically and Creatively.* Hillsdale, NJ, Lawrence Erlbaum Associates.

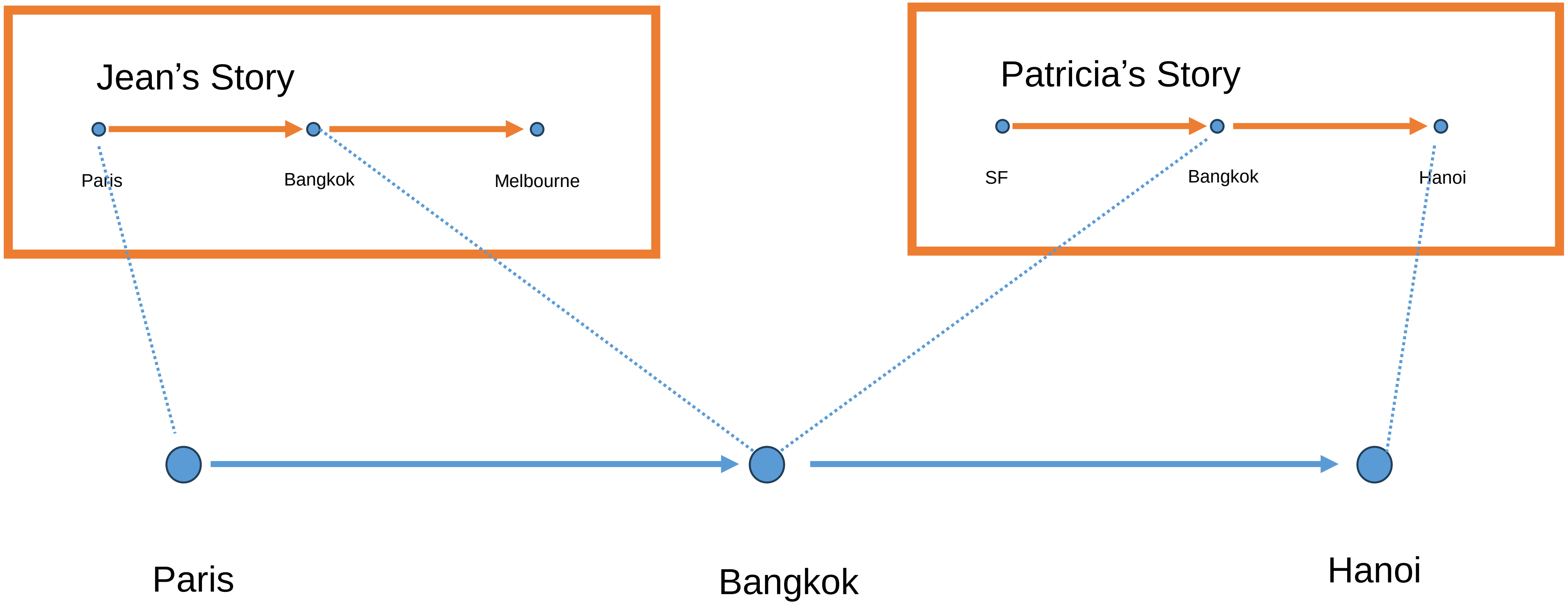


Fig. 1: Anne's Plan

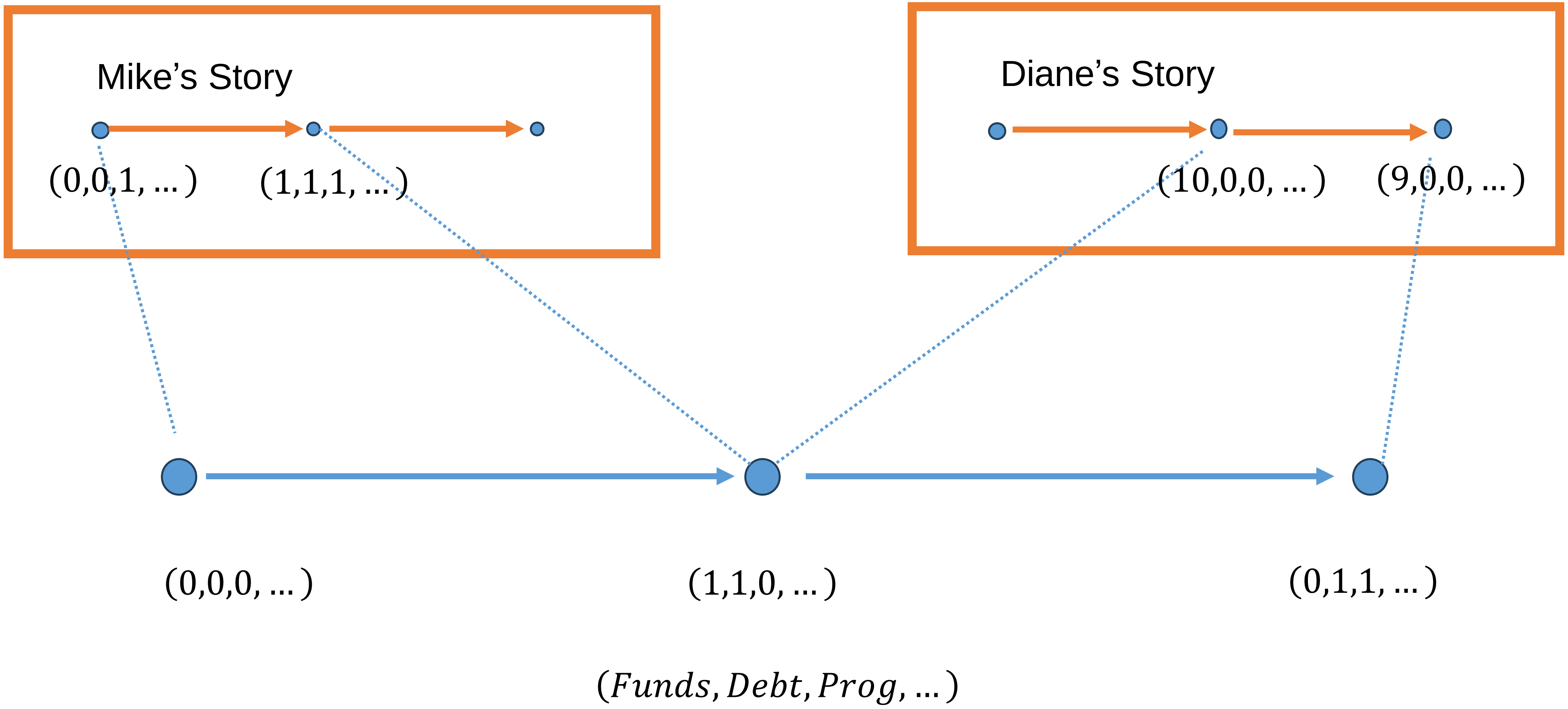
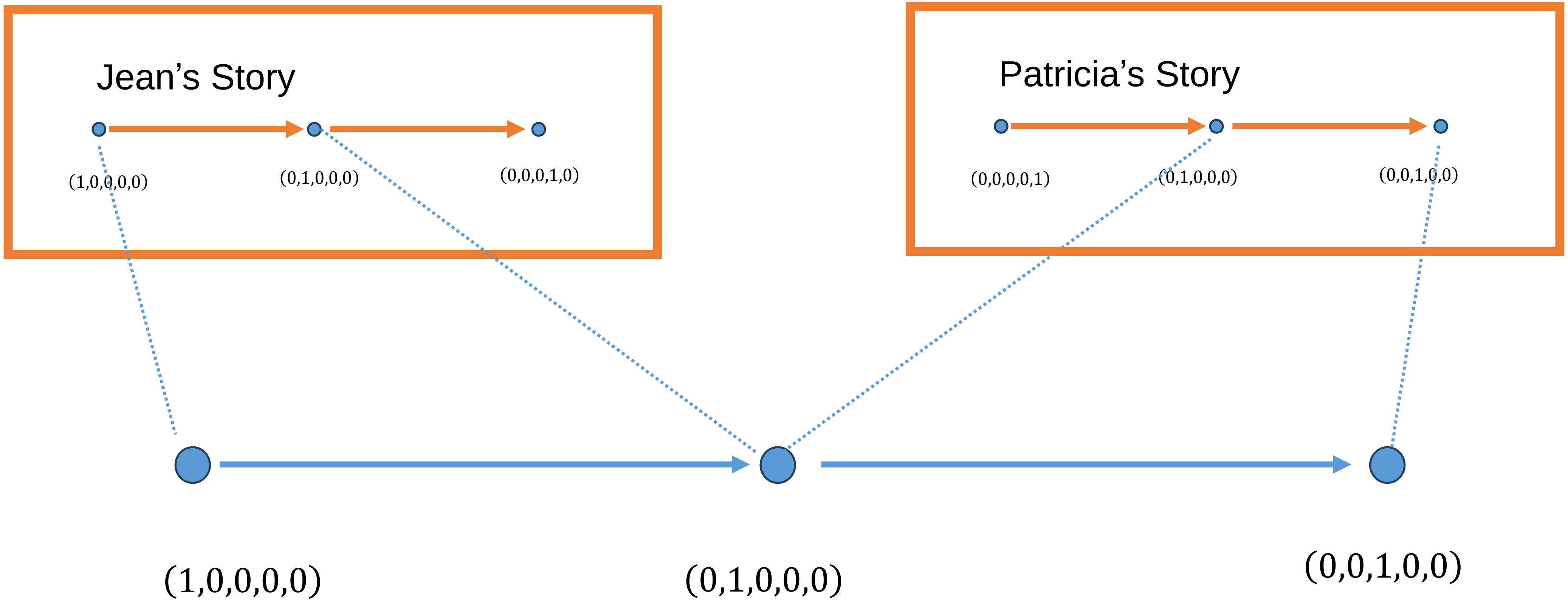


Fig. 2: Claire's Plan



$(x^1, x^2, \ldots, x^5) - x^j$ = indicator for presence in city j

Fig. 3: Anne's Plan – formally