Issues in Decision Making under Uncertainty

Lecture Outline

Deterministic Constrained Optimization Problem

Making decisions in a rational way is a problem which can be mathematically formulated as an optimization problem. Generally, several conflicting goals must be taken into account simultaneously. A choice has to be made about which goals are formulated as constraints, and which goal is reflected by a cost function.

General Problem ($\mathcal{P}_G$)

$$\min_{u \in U^{\mathrm{ad}} \subset \mathbb{U}} J(u) \tag{1a}$$

subject to

$$\Theta(u) \in -C \subset \mathbb{V} \; . \tag{1b}$$

Duality theory for constrained optimization problems provides a way to analyze the sensitivity of the best achievable cost as a function of (given) constraint levels.

Deterministic Optimal Control Problem

In a deterministic setting, problems that involve systems evolving in time enter the realm of optimal control.

Dynamic Problem $(\mathcal{P}_D)$

$$\min_{(u_0,\ldots,u_{T-1},x_0,\ldots,x_T)} \sum_{t=0}^{T-1} L_t(x_t, u_t) + K(x_T) \tag{2a}$$

subject to

$$\begin{cases} x_0 = x_{\text{ini}} \ \text{ given} \ , \\ x_{t+1} = f_t(x_t, u_t) \ , \quad t = 0, \ldots, T-1 \ . \end{cases} \tag{2b}$$

There are at least two points of view on optimal control:

- Maximum Principle (variational approach),
- Dynamic Programming (state space approach).

Deterministic Optimal Control Problem

In a deterministic setting, problems that involve systems evolving in time enter the realm of optimal control.

Dynamic Problem $(\mathcal{P}_D)$

$$\min_{(u_0,\dots,u_{T-1},x_0,\dots,x_T)} \sum_{t=0}^{T-1} L_t(x_t, u_t) + K(x_T) \tag{2a}$$

subject to

$$\begin{cases} x_0 = x_{\mathrm{ini}} \ \text{ given} \ , \\ x_{t+1} = f_t(x_t, u_t) \ , \quad t = 0, \dots, T-1 \ . \end{cases} \tag{2b}$$

There are at least two points of view on optimal control:

- Maximum Principle (variational approach),
- Dynamic Programming (state space approach).

Attitudes when Facing Uncertainty

In general, when making decisions, one is faced with uncertainties which affect the cost function and the constraints. Let us mention two possibilities (among others) for mathematically formulating decision making problems under uncertainty.

Attitudes when Facing Uncertainty

In general, when making decisions, one is faced with uncertainties which affect the cost function and the constraints. Let us mention two possibilities (among others) for mathematically formulating decision making problems under uncertainty.

- **Worst case design**. We assume that uncertainties lie in a bounded subset. We consider the worst situation to be faced and try to make it as good as possible. In more mathematical terms, one has to minimize the maximal possible value that Nature can give to the cost (robust control).
- **Probabilistic approach**. Uncertainties are viewed as random variables following a priori probability laws. Then the cost to be minimized is the expectation of some performance index depending on those random variables and on decisions (stochastic programming, stochastic optimal control).

Attitudes when Facing Uncertainty

In general, when making decisions, one is faced with uncertainties which affect the cost function and the constraints. Let us mention two possibilities (among others) for mathematically formulating decision making problems under uncertainty.

- **Worst case design**. We assume that uncertainties lie in a bounded subset. We consider the worst situation to be faced and try to make it as good as possible. In more mathematical terms, one has to minimize the maximal possible value that Nature can give to the cost (robust control).
- **Probabilistic approach**. Uncertainties are viewed as random variables following a priori probability laws. Then the cost to be minimized is the expectation of some performance index depending on those random variables and on decisions (stochastic programming, stochastic optimal control).

Decisions in the Probabilistic Framework (1)

We consider optimization problems in the probabilistic approach.

The decisions in such problems usually become random variables defined on the underlying probability space $(\Omega, \mathcal{A}, \mathbb{P})$. As a matter of fact, decisions $\boldsymbol{U}$ may depend on the uncertainties $\boldsymbol{W}$ that affect the problem, and are therefore themselves random variables.

An easy case is when those decisions are deterministic, that is, constant functions of the uncertainties : $\boldsymbol{U}(\omega) = u \quad \forall \omega \in \Omega$. Such decisions are termed **open-loop**[1] decisions.

A typical example of this situation is the investment problem: a decision maker has to make an investment in one time facing an uncertain future, so that its decision results from an (optimal) trade-off between all possible outcomes of the noise.

[1] here and now decisions in the stochastic programming terminology

Decisions in the Probabilistic Framework (1)

We consider optimization problems in the probabilistic approach.

The decisions in such problems usually become random variables defined on the underlying probability space $(\Omega, \mathcal{A}, \mathbb{P})$. As a matter of fact, decisions $\boldsymbol{U}$ may depend on the uncertainties $\boldsymbol{W}$ that affect the problem, and are therefore themselves random variables.

An easy case is when those decisions are deterministic, that is, constant functions of the uncertainties : $\boldsymbol{U}(\omega) = u \quad \forall \omega \in \Omega$. Such decisions are termed **open-loop**[1] decisions.

A typical example of this situation is the investment problem: a decision maker has to make an investment in one time facing an uncertain future, so that its decision results from an (optimal) trade-off between all possible outcomes of the noise.

[1]here and now decisions in the stochastic programming terminology

Decisions in the Probabilistic Framework (1)

We consider optimization problems in the probabilistic approach.

The decisions in such problems usually become random variables defined on the underlying probability space $(\Omega, \mathcal{A}, \mathbb{P})$. As a matter of fact, decisions $\boldsymbol{U}$ may depend on the uncertainties $\boldsymbol{W}$ that affect the problem, and are therefore themselves random variables.

An easy case is when those decisions are deterministic, that is, constant functions of the uncertainties : $\boldsymbol{U}(\omega) = u \quad \forall \omega \in \Omega$. Such decisions are termed **open-loop**[1] decisions.

A typical example of this situation is the investment problem: a decision maker has to make an investment in one time facing an uncertain future, so that its decision results from an (optimal) trade-off between all possible outcomes of the noise.

[1] here and now decisions in the stochastic programming terminology

Decisions in the Probabilistic Framework (2)

But the decisions may also be true random variables because they are produced by applying functions to the uncertainties. Here, we enter the domain of **closed-loop** (feedback) decisions $\boldsymbol{U} = \varphi(\boldsymbol{W})$, which plays a prominent part in optimization under uncertainty.

A well studied example of this situation is the so-called two-stage recourse problem. Here a decision maker takes a first decision u_0 (e.g. investment), after which a random event $\boldsymbol{W}_1$ occurs. Then a second recourse decision $\boldsymbol{U}_1$ (e.g. operation) is made assuming the noise known, that corrects the result of the first-stage decision.

More generally, we may consider multi-stage problems, for which a decision has to be taken at each time step of a given time horizon.

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \rightsquigarrow \boldsymbol{U}_2 \quad \ldots \quad \boldsymbol{W}_{T-1} \rightsquigarrow \boldsymbol{U}_{T-1} \rightsquigarrow \boldsymbol{W}_T$$

Decisions in the Probabilistic Framework (2)

But the decisions may also be true random variables because they are produced by applying functions to the uncertainties. Here, we enter the domain of **closed-loop** (feedback) decisions $\boldsymbol{U} = \varphi(\boldsymbol{W})$, which plays a prominent part in optimization under uncertainty.

A well studied example of this situation is the so-called two-stage recourse problem. Here a decision maker takes a first decision u_0 (e.g. investment), after which a random event $\boldsymbol{W}_1$ occurs. Then a second recourse decision $\boldsymbol{U}_1$ (e.g. operation) is made assuming the noise known, that corrects the result of the first-stage decision.

More generally, we may consider multi-stage problems, for which a decision has to be taken at each time step of a given time horizon.

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \rightsquigarrow \boldsymbol{U}_2 \quad \ldots \quad \boldsymbol{W}_{T-1} \rightsquigarrow \boldsymbol{U}_{T-1} \rightsquigarrow \boldsymbol{W}_T$$

Decisions in the Probabilistic Framework (2)

But the decisions may also be true random variables because they are produced by applying functions to the uncertainties. Here, we enter the domain of **closed-loop** (feedback) decisions $\boldsymbol{U} = \varphi(\boldsymbol{W})$, which plays a prominent part in optimization under uncertainty.

A well studied example of this situation is the so-called two-stage recourse problem. Here a decision maker takes a first decision u_0 *(e.g. investment), after which a random event* $\boldsymbol{W}_1$ *occurs. Then a second recourse decision* $\boldsymbol{U}_1$ *(e.g. operation) is made assuming the noise known, that corrects the result of the first-stage decision.*

More generally, we may consider multi-stage problems, for which a decision has to be taken at each time step of a given time horizon.

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \rightsquigarrow \boldsymbol{U}_2 \quad \ldots \quad \boldsymbol{W}_{T-1} \rightsquigarrow \boldsymbol{U}_{T-1} \rightsquigarrow \boldsymbol{W}_T$$

Stochastic Programming and Stochastic Control

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \quad \ldots \quad \rightsquigarrow \boldsymbol{U}_{t-1} \rightsquigarrow \boldsymbol{W}_t \rightsquigarrow \boldsymbol{U}_t \rightsquigarrow \boldsymbol{W}_{t+1} \quad \ldots$$

Stochastic Programming and Stochastic Control

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \ \ldots \ \rightsquigarrow \boldsymbol{U}_{t-1} \rightsquigarrow \boldsymbol{W}_t \rightsquigarrow \boldsymbol{U}_t \rightsquigarrow \boldsymbol{W}_{t+1} \ \ldots$$

- **Stochastic Programming** (SP) is the natural extension of Mathematical Programming to the stochastic framework. As such, numerical resolution methods are based on variational techniques. SP deals with static, two-stage and multi-stage problems. The question of information structure pops up in the field with multi-stage problems, at least to handle the constraints of nonanticipativeness: $\boldsymbol{U}_t = \varphi_t(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_t)$.
- **Stochastic Optimal Control** (SOC) is the extension of the theory of deterministic optimal control to the situation when uncertainties are present and modeled by random variables. SOC deals with dynamic problems. The standard resolution approach is Dynamic Programming (DP), which naturally delivers optimal feedbacks (as functions of the state).

Stochastic Programming and Stochastic Control

$$u_0 \rightsquigarrow \boldsymbol{W}_1 \rightsquigarrow \boldsymbol{U}_1 \rightsquigarrow \boldsymbol{W}_2 \ \ \ldots \ \ \rightsquigarrow \boldsymbol{U}_{t-1} \rightsquigarrow \boldsymbol{W}_t \rightsquigarrow \boldsymbol{U}_t \rightsquigarrow \boldsymbol{W}_{t+1} \ \ \ldots$$

- **Stochastic Programming** (SP) is the natural extension of Mathematical Programming to the stochastic framework. As such, numerical resolution methods are based on variational techniques. SP deals with static, two-stage and multi-stage problems. The question of information structure pops up in the field with multi-stage problems, at least to handle the constraints of nonanticipativeness: $\boldsymbol{U}_t = \varphi_t(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_t)$.
- **Stochastic Optimal Control** (SOC) is the extension of the theory of deterministic optimal control to the situation when uncertainties are present and modeled by random variables. SOC deals with dynamic problems. The standard resolution approach is Dynamic Programming (DP), which naturally delivers optimal feedbacks (as functions of the state).

Uncertainty and Information (1)

In a deterministic environnement, everything is known in advance!!!

In an uncertain environnement, the situation is quite different since trajectories are not predictable in advance because they depend on the realizations of random variables. Available observations of the noise reveal some information about those realizations. Using this information, one can do better than applying a naive open-loop control, as illustrated by the following example:

$$\min_{\boldsymbol{U}=u} \mathbb{E}\left(\left(\boldsymbol{U}-\boldsymbol{W}\right)^2\right) \qquad \text{versus} \qquad \min_{\boldsymbol{U}\in\mathcal{L}^0(\Omega,\mathbb{R})} \mathbb{E}\left(\left(\boldsymbol{U}-\boldsymbol{W}\right)^2\right) .$$

The achievable performance depends on the information pattern (information structure) of the problem: an optimization problem under uncertainty is not well-posed until the exact amount of information available prior to making a decision has been defined.

Uncertainty and Information (1)

In a deterministic environnement, everything is known in advance!!!

In an uncertain environnement, the situation is quite different since trajectories are not predictable in advance because they depend on the realizations of random variables. Available observations of the noise reveal some information about those realizations. Using this information, one can do better than applying a naive open-loop control, as illustrated by the following example:

$$\min_{\boldsymbol{U}=u} \mathbb{E}\Big(\big(\boldsymbol{U}-\boldsymbol{W}\big)^2\Big) \qquad \text{versus} \qquad \min_{\boldsymbol{U}\in\mathcal{L}^0(\Omega,\mathbb{R})} \mathbb{E}\Big(\big(\boldsymbol{U}-\boldsymbol{W}\big)^2\Big) .$$

The achievable performance depends on the information pattern (information structure) of the problem: an optimization problem under uncertainty is not well-posed until the exact amount of information available prior to making a decision has been defined.

Uncertainty and Information (1)

In a deterministic environnement, everything is known in advance!!!

In an uncertain environnement, the situation is quite different since trajectories are not predictable in advance because they depend on the realizations of random variables. Available observations of the noise reveal some information about those realizations. Using this information, one can do better than applying a naive open-loop control, as illustrated by the following example:

$$\min_{\boldsymbol{U}=u} \mathbb{E}\Big(\big(\boldsymbol{U} - \boldsymbol{W}\big)^2\Big) \qquad \text{versus} \qquad \min_{\boldsymbol{U}\in\mathcal{L}^0(\Omega,\mathbb{R})} \mathbb{E}\Big(\big(\boldsymbol{U} - \boldsymbol{W}\big)^2\Big) .$$

The achievable performance depends on the information pattern (information structure) of the problem: an optimization problem under uncertainty is **not well-posed** until the exact amount of information available prior to making a decision has been defined.

Uncertainty and Information (2)

Solving stochastic optimization problems is not just a matter of optimization: it is also the question of handling specific constraints representing the information structure. There are essentially two ways of dealing with such constraints.

A tricky aspect of information patterns is that future information may be affected by past decisions, leading to the so-called **dual effect**. *Indeed a decision has two objectives: contributing to optimizing the cost function, and modifying the information structure for future decisions. Problems with dual effect are generally among the most difficult.*

Uncertainty and Information (2)

Solving stochastic optimization problems is not just a matter of optimization: it is also the question of handling specific constraints representing the information structure. There are essentially two ways of dealing with such constraints.

- That used by the DP approach is a functional way: decisions are searched for as functions of observations (feedback).
- Another way is to consider all variables as random variables: then the information constraints are expressed by the notion of measurability, mathematically captured by σ-algebras.

A tricky aspect of information patterns is that future information may be affected by past decisions, leading to the so-called **dual effect**. *Indeed a decision has two objectives: contributing to optimizing the cost function, and modifying the information structure for future decisions. Problems with dual effect are generally among the most difficult.*

Uncertainty and Information (2)

Solving stochastic optimization problems is not just a matter of optimization: it is also the question of handling specific constraints representing the information structure. There are essentially two ways of dealing with such constraints.

- That used by the DP approach is a functional way: decisions are searched for as functions of observations (feedback).
- Another way is to consider all variables as random variables: then the information constraints are expressed by the notion of measurability, mathematically captured by σ-algebras.

A tricky aspect of information patterns is that future information may be affected by past decisions, leading to the so-called dual effect. Indeed a decision has two objectives: contributing to optimizing the cost function, and modifying the information structure for future decisions. Problems with dual effect are generally among the most difficult.

Uncertainty and Information (2)

Solving stochastic optimization problems is not just a matter of optimization: it is also the question of handling specific constraints representing the information structure. There are essentially two ways of dealing with such constraints.

- That used by the DP approach is a functional way: decisions are searched for as functions of observations (feedback).
- Another way is to consider all variables as random variables: then the information constraints are expressed by the notion of measurability, mathematically captured by σ-algebras.

A tricky aspect of information patterns is that future information may be affected by past decisions, leading to the so-called dual effect. Indeed a decision has two objectives: contributing to optimizing the cost function, and modifying the information structure for future decisions. Problems with dual effect are generally among the most difficult.

Uncertainty and Information (2)

Solving stochastic optimization problems is not just a matter of optimization: it is also the question of handling specific constraints representing the information structure. There are essentially two ways of dealing with such constraints.

- That used by the DP approach is a functional way: decisions are searched for as functions of observations (feedback).
- Another way is to consider all variables as random variables: then the information constraints are expressed by the notion of measurability, mathematically captured by σ-algebras.

A tricky aspect of information patterns is that future information may be affected by past decisions, leading to the so-called **dual effect**. *Indeed a decision has two objectives: contributing to optimizing the cost function, and modifying the information structure for future decisions. Problems with dual effect are generally among the most difficult.*

Let's Summarize

In summary, solving stochastic optimization problems is not only a matter of optimizing a criterion under conventional constraints. Issues and expected difficulties are the following.

1. How to compute mathematical expectations?
    - generally a difficult task by itself...
2. How to formulate and how to deal with constraints:
    - in an almost-sure sense?
    - in expectation?
    - in probability?
3. How to properly handle informational constraints?

Extension of the General Problem $(\mathcal{P}_G)$ (1)

We start from Problem (1), and assume now that the control is a random variable $\boldsymbol{U}$ defined on the probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on $(\mathbb{U}, \mathcal{U})$. The cost j is affected by a noise $\boldsymbol{W}$:[2]

$$J(\boldsymbol{U}) = \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) .$$

Denote by $\mathcal{F}$ the σ-field generated by $\boldsymbol{W}$. The *(interesting part of)* information available to the decision maker is a piece of the information revealed by the noise $\boldsymbol{W}$, and thus is represented by a σ-field $\mathcal{G}$ included in $\mathcal{F}$. Then the optimization problem writes

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) ,$$

where $\boldsymbol{U} \preceq \mathcal{G}$ means that $\boldsymbol{U}$ is measurable w.r.t. the σ-field $\mathcal{G}$.

[2] There is here a tricky point in the notations. . .

Extension of the General Problem $(\mathcal{P}_G)$ (1)

We start from Problem (1), and assume now that the control is a random variable $\boldsymbol{U}$ defined on the probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on $(\mathbb{U}, \mathcal{U})$. The cost j is affected by a noise $\boldsymbol{W}$:[2]

$$J(\boldsymbol{U}) = \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) .$$

Denote by $\mathcal{F}$ the σ-field generated by $\boldsymbol{W}$. The *(interesting part of)* information available to the decision maker is a piece of the information revealed by the noise $\boldsymbol{W}$, and thus is represented by a σ-field $\mathcal{G}$ included in $\mathcal{F}$. Then the optimization problem writes

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) ,$$

where $\boldsymbol{U} \preceq \mathcal{G}$ means that $\boldsymbol{U}$ is measurable w.r.t. the σ-field $\mathcal{G}$.

[2] There is here a tricky point in the notations. . .

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) \ .$$

Examples.

These three examples correspond to a static information structure, that is, the case where the σ–field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) \;.$$

Examples.

1. $\mathcal{G} = \{\emptyset, \Omega\}$: this corresponds to the open-loop case:

$$\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) \;.$$

2. $\mathcal{G} \supset \sigma(\boldsymbol{W})$: we have by the interchange theorem:

$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} j(u, w)\Big) \;, \quad \forall w \in \mathbb{W} \;.$$

3. $\mathcal{G} \subset \sigma(\boldsymbol{W})$: the problem is equivalent to

$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W}) \mid \mathcal{G}\big)\Big) \;.$$

These three examples correspond to a static information structure, that is, the case where the σ-field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) \; .$$

Examples.

1. $\mathcal{G} = \{\emptyset, \Omega\}$: this corresponds to the open-loop case:

$$\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) \; .$$

2. $\mathcal{G} \supset \sigma\big(\boldsymbol{W}\big)$: we have by the interchange theorem:

$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} j(u, w)\Big) \; , \quad \forall w \in \mathbb{W} \; .$$

3. $\mathcal{G} \subset \sigma\big(\boldsymbol{W}\big)$: the problem is equivalent to

$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W}) \mid \mathcal{G}\big)\Big) \; .$$

These three examples correspond to a static information structure, that is, the case where the σ-field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) .$$

Examples.

1. $\mathcal{G} = \{\emptyset, \Omega\}$: this corresponds to the open-loop case:
$$\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$
2. $\mathcal{G} \supset \sigma(\boldsymbol{W})$: we have by the interchange theorem:
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} j(u, w)\Big) , \quad \forall w \in \mathbb{W} .$$
3. $\mathcal{G} \subset \sigma(\boldsymbol{W})$: the problem is equivalent to
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W}) \mid \mathcal{G}\big)\Big) .$$

These three examples correspond to a static information structure, that is, the case where the σ-field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) \ .$$

Examples.

1. $\mathcal{G} = \{\emptyset, \Omega\}$: this corresponds to the open-loop case:
$$\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) \ .$$
2. $\mathcal{G} \supset \sigma\big(\boldsymbol{W}\big)$: we have by the interchange theorem:
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} j(u, w)\Big) \ , \quad \forall w \in \mathbb{W} \ .$$
3. $\mathcal{G} \subset \sigma\big(\boldsymbol{W}\big)$: the problem is equivalent to
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W}) \mid \mathcal{G}\big)\Big) \ .$$

These three examples correspond to a static information structure, that is, the case where the σ-field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (2)

$$\min_{\boldsymbol{U} \preceq \mathcal{G}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) .$$

Examples.

1. $\mathcal{G} = \{\emptyset, \Omega\}$: this corresponds to the open-loop case:
$$\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$
2. $\mathcal{G} \supset \sigma\big(\boldsymbol{W}\big)$: we have by the interchange theorem:
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} j(u, w)\Big) , \quad \forall w \in \mathbb{W} .$$
3. $\mathcal{G} \subset \sigma\big(\boldsymbol{W}\big)$: the problem is equivalent to
$$\mathbb{E}\Big(\min_{u \in \mathbb{U}} \mathbb{E}\big(j(u, \boldsymbol{W}) \bigm| \mathcal{G}\big)\Big) .$$

These three examples correspond to a static information structure, that is, the case where the σ-field $\mathcal{G}$ is fixed.

Extension of the General Problem $(\mathcal{P}_G)$ (3)

When time is involved in the problem, the decision $\boldsymbol{U}$ rewrites as a collection $(\boldsymbol{U}_0, \ldots, \boldsymbol{U}_{T-1})$ of random variables, each subject to its own measurability constraint:

$$\boldsymbol{U}_t \preceq \mathcal{G}_t \; .$$

The problem is dynamic (time), but involves a static information structure as soon as each σ-field $\mathcal{G}_t$ is fixed, that is, the information available at time t is not modified by past controls. This is surely the case when dealing with non anticipativy constraints:

$$\mathcal{G}_t = \sigma(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_t) \; .$$

The situation of dynamic information structure occurs when $\mathcal{G}_t$ depends on past controls, as in $\mathcal{G}_1 = \sigma(\boldsymbol{W}_1)$, $\mathcal{G}_2 = \sigma(\boldsymbol{U}_1, \boldsymbol{W}_2)$.

Extension of the General Problem $(\mathcal{P}_G)$ (3)

When time is involved in the problem, the decision $\boldsymbol{U}$ rewrites as a collection $(\boldsymbol{U}_0, \ldots, \boldsymbol{U}_{T-1})$ of random variables, each subject to its own measurability constraint:

$$\boldsymbol{U}_t \preceq \mathcal{G}_t \ .$$

The problem is dynamic (time), but involves a static information structure as soon as each σ-field $\mathcal{G}_t$ is fixed, that is, the information available at time t is not modified by past controls. This is surely the case when dealing with non anticipativy constraints:

$$\mathcal{G}_t = \sigma(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_t) \ .$$

The situation of dynamic information structure occurs when $\mathcal{G}_t$ depends on past controls, as in $\mathcal{G}_1 = \sigma(\boldsymbol{W}_1)$, $\mathcal{G}_2 = \sigma(\boldsymbol{U}_1, \boldsymbol{W}_2)$.

Extension of the General Problem $(\mathcal{P}_G)$ (3)

When time is involved in the problem, the decision $\boldsymbol{U}$ rewrites as a collection $(\boldsymbol{U}_0, \ldots, \boldsymbol{U}_{T-1})$ of random variables, each subject to its own measurability constraint:

$$\boldsymbol{U}_t \preceq \mathcal{G}_t \, .$$

The problem is dynamic (time), but involves a static information structure as soon as each σ-field $\mathcal{G}_t$ is fixed, that is, the information available at time t is not modified by past controls. This is surely the case when dealing with non anticipativy constraints:

$$\mathcal{G}_t = \sigma(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_t) \, .$$

The situation of dynamic information structure occurs when $\mathcal{G}_t$ depends on past controls, as in $\mathcal{G}_1 = \sigma(\boldsymbol{W}_1)$, $\mathcal{G}_2 = \sigma(\boldsymbol{U}_1, \boldsymbol{W}_2)$.

Extension of the General Problem $(\mathcal{P}_G)$ (4)

Generally, the σ-field $\mathcal{G}$ is generated by an observation, that is, a random variable $\boldsymbol{Y}$: $\mathcal{G} = \sigma\big(\boldsymbol{Y}\big)$. Then, the information constraint writes $\boldsymbol{U} \preceq \boldsymbol{Y}$, and the optimization problem is:

$$\min_{\boldsymbol{U} \preceq \boldsymbol{Y}} \mathbb{E}\big(j(\boldsymbol{U}, \boldsymbol{W})\big) \ .$$

In this setting, a static information structure may correspond to an observation depending only on the noise: $\boldsymbol{Y} = g\big(\boldsymbol{W}\big)$, and a dynamic information structure to an observation depending on both the control and the noise: $\boldsymbol{Y} = g\big(\boldsymbol{U}, \boldsymbol{W}\big)$.

Extension of the General Problem $(\mathcal{P}_G)$ (4)

Generally, the σ-field $\mathcal{G}$ is generated by an observation, that is, a random variable $\boldsymbol{Y}$: $\mathcal{G} = \sigma\big(\boldsymbol{Y}\big)$. Then, the information constraint writes $\boldsymbol{U} \preceq \boldsymbol{Y}$, and the optimization problem is:

$$\min_{\boldsymbol{U} \preceq \boldsymbol{Y}} \mathbb{E}\left(j(\boldsymbol{U}, \boldsymbol{W})\right) .$$

In this setting, a static information structure may correspond to an observation depending only on the noise: $\boldsymbol{Y} = g(\boldsymbol{W})$, and a dynamic information structure to an observation depending on both the control and the noise: $\boldsymbol{Y} = g(\boldsymbol{U}, \boldsymbol{W})$.

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (1)

The natural stochastic extension of Problem (2) consists in adding a perturbation $\boldsymbol{W}_t$ at each time step t:

$$\min_{(\boldsymbol{U}_0,\ldots,\boldsymbol{U}_{T-1},\boldsymbol{X}_0,\ldots,\boldsymbol{X}_T)} \mathbb{E}\left(\sum_{t=0}^{T-1} L_t(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{W}_{t+1}) + K(\boldsymbol{X}_T) \right)$$

subject to:

$$\begin{cases} \boldsymbol{X}_0 &= f_{-1}(\boldsymbol{W}_0) \ , \\ \boldsymbol{X}_{t+1} &= f_t(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{W}_{t+1}) \ , \quad t = 0, \ldots, T-1 \ . \end{cases}$$

Problem not well-posed: the information structure not defined!

We denote by $\mathcal{F}_t$ the σ-field generated by noises prior time t:

$$\mathcal{F}_t = \sigma\left(\boldsymbol{W}_0, \ldots, \boldsymbol{W}_t\right) \ , \quad t = 0, \ldots, T \ .$$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (1)

The natural stochastic extension of Problem (2) consists in adding a perturbation $\boldsymbol{W}_t$ at each time step t:

$$\min_{(\boldsymbol{U}_0,\dots,\boldsymbol{U}_{T-1},\boldsymbol{X}_0,\dots,\boldsymbol{X}_T)} \mathbb{E}\left(\sum_{t=0}^{T-1} L_t(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{W}_{t+1}) + K(\boldsymbol{X}_T) \right)$$

subject to:

$$\begin{cases} \boldsymbol{X}_0 & = f_{-1}(\boldsymbol{W}_0) \ , \\ \boldsymbol{X}_{t+1} & = f_t(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{W}_{t+1}) \ , \quad t = 0, \dots, T-1 \ . \end{cases}$$

Problem not well-posed: the information structure not defined!

We denote by $\mathcal{F}_t$ the σ-field generated by noises prior time t:

$$\mathcal{F}_t = \sigma\big(\boldsymbol{W}_0, \dots, \boldsymbol{W}_t\big) \ , \quad t = 0, \dots, T \ .$$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (2)

Information Structure.

- A new observation becomes available at each time t:

$$Z_t = h_t(X_t, W_t) , \quad t = 0, \ldots, T-1 .$$

  - $Z_t = W_t$: observation of the noise,
  - $Z_t = X_t$: observation of the state.

- Information at time t is a function of past observations:

$$Y_t = C_t(Z_0, \ldots, Z_t) , \quad t = 0, \ldots, T-1 .$$

  - $Y_t = Z_t$: memoryless information,
  - $Y_t = (Z_0, \ldots, Z_t)$: perfect memory.

Information constraints: $U_t \preceq Y_t$, $t = 0, \ldots, T-1$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (2)

Information Structure.

- A new observation becomes available at each time t:

$$\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \ , \quad t = 0, \ldots, T-1 \ .$$

  - $\boldsymbol{Z}_t = \boldsymbol{W}_t$: observation of the noise,
  - $\boldsymbol{Z}_t = \boldsymbol{X}_t$: observation of the state.

- Information at time t is a function of past observations:

$$\boldsymbol{Y}_t = C_t(\boldsymbol{Z}_0, \ldots, \boldsymbol{Z}_t) \ , \quad t = 0, \ldots, T-1 \ .$$

  - $\boldsymbol{Y}_t = \boldsymbol{Z}_t$: memoryless information,
  - $\boldsymbol{Y}_t = (\boldsymbol{Z}_0, \ldots, \boldsymbol{Z}_t)$: perfect memory.

Information constraints: $\boldsymbol{U}_t \preceq \boldsymbol{Y}_t \ , \quad t = 0, \ldots, T-1$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (2)

Information Structure.

- A new observation becomes available at each time t:

$$\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \ , \quad t = 0, \dots, T-1 \ .$$

  - $\boldsymbol{Z}_t = \boldsymbol{W}_t$: observation of the noise,
  - $\boldsymbol{Z}_t = \boldsymbol{X}_t$: observation of the state.

- Information at time t is a function of past observations:

$$\boldsymbol{Y}_t = C_t\big(\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t\big) \ , \quad t = 0, \dots, T-1 \ .$$

  - $\boldsymbol{Y}_t = \boldsymbol{Z}_t$: memoryless information.
  - $\boldsymbol{Y}_t = (\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$: perfect memory.

Information constraints: $\boldsymbol{U}_t \preceq \boldsymbol{Y}_t \ , \quad t = 0, \dots, T-1$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (2)

Information Structure.

- A new observation becomes available at each time t:

$$\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \; , \quad t = 0, \ldots, T-1 \; .$$

  - $\boldsymbol{Z}_t = \boldsymbol{W}_t$: observation of the noise,
  - $\boldsymbol{Z}_t = \boldsymbol{X}_t$: observation of the state.

- Information at time t is a function of past observations:

$$\boldsymbol{Y}_t = C_t\big(\boldsymbol{Z}_0, \ldots, \boldsymbol{Z}_t\big) \; , \quad t = 0, \ldots, T-1 \; .$$

  - $\boldsymbol{Y}_t = \boldsymbol{Z}_t$: memoryless information.
  - $\boldsymbol{Y}_t = (\boldsymbol{Z}_0, \ldots, \boldsymbol{Z}_t)$: perfect memory.

Information constraints: $\boldsymbol{U}_t \preceq \boldsymbol{Y}_t \; , \quad t = 0, \ldots, T-1$

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (3)

Functional Approach: Stochastic Optimal Control

$\rightsquigarrow$ Assumptions on the noise process $\{\boldsymbol{W}_t\}_{t=0,\ldots,T}$.

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (3)

Functional Approach: Stochastic Optimal Control

$\leadsto$ Assumptions on the noise process $\{\boldsymbol{W}_t\}_{t=0,\dots,T}$.

- **Markovian case:** $\boldsymbol{Z}_t = \boldsymbol{X}_t \;\; / \;\; \boldsymbol{Y}_t \succeq \boldsymbol{Z}_t$.
  Solution may be computed by the Dynamic Programming approach, developed on the state $\boldsymbol{X}_t$: $\boldsymbol{U}_t = \varphi_t(\boldsymbol{X}_t)$.
  $\leadsto$ Curse of dimensionality.
- **Classical case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \;\; / \;\; \boldsymbol{Y}_t = (\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$.
  The Dynamic Programming approach is still available, the state being the probability law of $\boldsymbol{X}_t$ rather than $\boldsymbol{X}_t$ itself.
- **General case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \;\; / \;\; \boldsymbol{Y}_t = C_t(\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$.
  We are usually not able to solve the optimality conditions (dual effect, Witsenhausen counterexample).

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (3)

Functional Approach: Stochastic Optimal Control

$\rightsquigarrow$ Assumptions on the noise process $\{\boldsymbol{W}_t\}_{t=0,\dots,T}$.

- **Markovian case:** $\boldsymbol{Z}_t = \boldsymbol{X}_t \ / \ \boldsymbol{Y}_t \succeq \boldsymbol{Z}_t$.
  Solution may be computed by the Dynamic Programming approach, developed on the state $\boldsymbol{X}_t$: $\boldsymbol{U}_t = \varphi_t(\boldsymbol{X}_t)$.
  $\rightsquigarrow$ Curse of dimensionality.
- **Classical case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \ / \ \boldsymbol{Y}_t = (\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$.
  The Dynamic Programming approach is still available, the state being the probability law of $\boldsymbol{X}_t$ rather than $\boldsymbol{X}_t$ itself.
- **General case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \ / \ \boldsymbol{Y}_t = C_t(\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$.
  We are usually not able to solve the optimality conditions (dual effect, Witsenhausen counterexample).

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (3)

Functional Approach: Stochastic Optimal Control

⇝ Assumptions on the noise process $\{\boldsymbol{W}_t\}_{t=0,\dots,T}$.

- **Markovian case:** $\boldsymbol{Z}_t = \boldsymbol{X}_t \;\; / \;\; \boldsymbol{Y}_t \succeq \boldsymbol{Z}_t$.
  Solution may be computed by the Dynamic Programming approach, developed on the state $\boldsymbol{X}_t$: $\boldsymbol{U}_t = \varphi_t\big(\boldsymbol{X}_t\big)$.
  ⇝ Curse of dimensionality.
- **Classical case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \;\; / \;\; \boldsymbol{Y}_t = (\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t)$.
  The Dynamic Programming approach is still available, the state being the probability law of $\boldsymbol{X}_t$ rather than $\boldsymbol{X}_t$ itself.
- **General case:** $\boldsymbol{Z}_t = h_t(\boldsymbol{X}_t, \boldsymbol{W}_t) \;\; / \;\; \boldsymbol{Y}_t = C_t\big(\boldsymbol{Z}_0, \dots, \boldsymbol{Z}_t\big)$.
  We are usually not able to solve the optimality conditions (dual effect, Witsenhausen counterexample).

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (4)

Some remarks on the Markovian case

We assume that the noise process is a white noise, that is, the random variables $\{\boldsymbol{W}_t\}_{t=0,\ldots,T}$ are independent of each other.

The Markovian case is the situation when the information $\boldsymbol{Y}_t$ available at time t is a perfect observation of the state $\boldsymbol{X}_t$. If the observation is partial or noisy, the Markovian situation is broken.

Note that, in the Markovian case, the information does depend, in general, upon past controls $\{\boldsymbol{U}_s\}_{s<t}$, hence dual effect.
But we would not do better replacing $\sigma(\boldsymbol{Y}_t)$ by $\sigma(\boldsymbol{W}_0, \ldots, \boldsymbol{W}_t)$!
The Markovian case, although falling into the category of problems with a dual effect, is in fact not so complex...

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (4)

Some remarks on the Markovian case

We assume that the noise process is a white noise, that is, the random variables $\{\boldsymbol{W}_t\}_{t=0,\ldots,T}$ are independent of each other.

The Markovian case is the situation when the information $\boldsymbol{Y}_t$ available at time t is a perfect observation of the state $\boldsymbol{X}_t$. If the observation is partial or noisy, the Markovian situation is broken.

Note that, in the Markovian case, the information does depend, in general, upon past controls $\{\boldsymbol{U}_s\}_{s<t}$, hence dual effect.
But we would not do better replacing $\sigma(\boldsymbol{Y}_t)$ by $\sigma(\boldsymbol{W}_0, \ldots, \boldsymbol{W}_t)$!
The Markovian case, although falling into the category of problems with a dual effect, is in fact not so complex...

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (4)

Some remarks on the Markovian case

We assume that the noise process is a white noise, that is, the random variables $\{\boldsymbol{W}_t\}_{t=0,\dots,T}$ are independent of each other.

The Markovian case is the situation when the information $\boldsymbol{Y}_t$ available at time t is a perfect observation of the state $\boldsymbol{X}_t$. If the observation is partial or noisy, the Markovian situation is broken.

Note that, in the Markovian case, the information does depend, in general, upon past controls $\{\boldsymbol{U}_s\}_{s<t}$, hence dual effect.

But we would not do better replacing $\sigma(\boldsymbol{Y}_t)$ by $\sigma(\boldsymbol{W}_0, \dots, \boldsymbol{W}_t)$!
The Markovian case, although falling into the category of problems with a dual effect, is in fact not so complex...

Extension of the Dynamic Problem $(\mathcal{P}_D)$ (4)

Some remarks on the Markovian case

We assume that the noise process is a white noise, that is, the random variables $\{\boldsymbol{W}_t\}_{t=0,\dots,T}$ are independent of each other.

The Markovian case is the situation when the information $\boldsymbol{Y}_t$ available at time t is a perfect observation of the state $\boldsymbol{X}_t$. If the observation is partial or noisy, the Markovian situation is broken.

Note that, in the Markovian case, the information does depend, in general, upon past controls $\{\boldsymbol{U}_s\}_{s<t}$, hence dual effect.
But we would not do better replacing $\sigma(\boldsymbol{Y}_t)$ by $\sigma(\boldsymbol{W}_0, \dots, \boldsymbol{W}_t)$!
The Markovian case, although falling into the category of problems with a dual effect, is in fact not so complex. . .

Witsenhausen's Couterexample (1)

This problem was proposed by **Hans Witsenhausen** in 1968 as evidence that a LQG problem may lead to nonlinear feedback solutions whenever the information structure is not classical.

$$\begin{aligned}
\min_{\boldsymbol{U}_0 \preceq \boldsymbol{Y}_0,\, \boldsymbol{U}_1 \preceq \boldsymbol{Y}_1} \quad & \mathbb{E}\big(\alpha \boldsymbol{U}_0^2 + \boldsymbol{X}_2^2\big) \\
\text{s.t.} \quad & \boldsymbol{X}_0 = \boldsymbol{W}_0 \,, \\
& \boldsymbol{X}_1 = \boldsymbol{X}_0 + \boldsymbol{U}_0 \,, \\
& \boldsymbol{X}_2 = \boldsymbol{X}_1 - \boldsymbol{U}_1 \,, \\
& \boldsymbol{Y}_0 = \boldsymbol{X}_0 \,, \\
& \boldsymbol{Y}_1 = \boldsymbol{X}_1 + \boldsymbol{W}_1 \,.
\end{aligned}$$

- The observation of the state is noisy!
- The perfect memory assumption is not satisfied!

The exact solution is so far unknown!

Witsenhausen's Couterexample (1)

This problem was proposed by **Hans Witsenhausen** in 1968 as evidence that a LQG problem may lead to nonlinear feedback solutions whenever the information structure is not classical.

$$
\begin{aligned}
\min_{\boldsymbol{U}_0 \preceq \boldsymbol{Y}_0, \boldsymbol{U}_1 \preceq \boldsymbol{Y}_1} \quad & \mathbb{E}\big(\alpha \boldsymbol{U}_0^2 + \boldsymbol{X}_2^2\big) \\
\text{s.t.} \quad & \boldsymbol{X}_0 = \boldsymbol{W}_0 , \\
& \boldsymbol{X}_1 = \boldsymbol{X}_0 + \boldsymbol{U}_0 , \\
& \boldsymbol{X}_2 = \boldsymbol{X}_1 - \boldsymbol{U}_1 , \\
& \boldsymbol{Y}_0 = \boldsymbol{X}_0 , \\
& \boldsymbol{Y}_1 = \boldsymbol{X}_1
\end{aligned}
$$

Witsenhausen's Couterexample (1)

This problem was proposed by **Hans Witsenhausen** in 1968 as evidence that a LQG problem may lead to nonlinear feedback solutions whenever the information structure is not classical.

$$\min_{\boldsymbol{U}_0 \preceq \boldsymbol{Y}_0, \boldsymbol{U}_1 \preceq \boldsymbol{Y}_1} \quad \mathbb{E}\big(\alpha \boldsymbol{U}_0^2 + \boldsymbol{X}_2^2\big)$$

$$\text{s.t.} \quad \boldsymbol{X}_0 = \boldsymbol{W}_0 \ ,$$

$$\boldsymbol{X}_1 = \boldsymbol{X}_0 + \boldsymbol{U}_0 \ ,$$

$$\boldsymbol{X}_2 = \boldsymbol{X}_1 - \boldsymbol{U}_1 \ ,$$

$$\boldsymbol{Y}_0 = \boldsymbol{X}_0 \ ,$$

$$\boldsymbol{Y}_1 = \boldsymbol{X}_1 + \boldsymbol{W}_1 \ .$$

- The observation of the state is noisy!
- The perfect memory assumption is not satisfied!

The exact solution is so far unknown!

Witsenhausen's Couterexample (1)

This problem was proposed by **Hans Witsenhausen** in 1968 as evidence that a LQG problem may lead to nonlinear feedback solutions whenever the information structure is not classical.

$$
\begin{aligned}
\min_{\boldsymbol{U}_0 \preceq \boldsymbol{Y}_0, \boldsymbol{U}_1 \preceq \boldsymbol{Y}_1} \quad & \mathbb{E}\big(\alpha \boldsymbol{U}_0^2 + \boldsymbol{X}_2^2\big) \\
\text{s.t.} \quad & \boldsymbol{X}_0 = \boldsymbol{W}_0 \ , \\
& \boldsymbol{X}_1 = \boldsymbol{X}_0 + \boldsymbol{U}_0 \ , \\
& \boldsymbol{X}_2 = \boldsymbol{X}_1 - \boldsymbol{U}_1 \ , \\
& \boldsymbol{Y}_0 = \boldsymbol{X}_0 \ , \\
& \boldsymbol{Y}_1 = \boldsymbol{X}_1 + \boldsymbol{W}_1 \ .
\end{aligned}
$$

- The observation of the state is noisy!
- The perfect memory assumption is not satisfied!

The exact solution is so far unknown!

Witsenhausen's Couterexample (1)

This problem was proposed by **Hans Witsenhausen** in 1968 as evidence that a LQG problem may lead to nonlinear feedback solutions whenever the information structure is not classical.

$$
\begin{aligned}
\min_{\boldsymbol{U}_0 \preceq \boldsymbol{Y}_0, \boldsymbol{U}_1 \preceq \boldsymbol{Y}_1} \quad & \mathbb{E}\big(\alpha \boldsymbol{U}_0^2 + \boldsymbol{X}_2^2\big) \\
\text{s.t.} \quad & \boldsymbol{X}_0 = \boldsymbol{W}_0 , \\
& \boldsymbol{X}_1 = \boldsymbol{X}_0 + \boldsymbol{U}_0 , \\
& \boldsymbol{X}_2 = \boldsymbol{X}_1 - \boldsymbol{U}_1 , \\
& \boldsymbol{Y}_0 = \boldsymbol{X}_0 , \\
& \boldsymbol{Y}_1 = \boldsymbol{X}_1 + \boldsymbol{W}_1 .
\end{aligned}
$$

- The observation of the state is noisy!
- The perfect memory assumption is not satisfied!

The exact solution is so far unknown!

Witsenhausen's Couterexample (2)

Let's try to give the feeling of how dual effect works.

In order to have the second cost term X_2^2 as close as possible to zero, we have to guess the value of W_0 at $t = 1$!
(indeed we have $X_2 = X_0 + U_0 - U_1$, with $X_0 = W_0$ and $U_0 \preceq W_0$).

- If we use a linear strategy at $t = 0$: $U_0 = (\kappa - 1) W_0$,
- then the information available at $t = 1$ is $Y_1 = \kappa W_0 + W_1$,
- so that if κ is big enough, $Y_1 / \kappa \approx W_0$:
- the decision maker at $t = 1$ may accurately know W_0.

But increasing κ increases the first cost term αU_0^2...

In the classical case $Y_0 = X_0$, $Y_1 = (X_0, X_1 + W_1)$ (perfect memory), the optimal solution is known: $(U_0^\sharp, U_1^\sharp) = (0, X_0)$.

Witsenhausen's Couterexample (2)

Let's try to give the feeling of how dual effect works.

In order to have the second cost term $\boldsymbol{X}_2^2$ as close as possible to zero, we have to guess the value of $\boldsymbol{W}_0$ at $t = 1$!
(indeed we have $\boldsymbol{X}_2 = \boldsymbol{X}_0 + \boldsymbol{U}_0 - \boldsymbol{U}_1$, with $\boldsymbol{X}_0 = \boldsymbol{W}_0$ and $\boldsymbol{U}_0 \preceq \boldsymbol{W}_0$).

- If we use a linear strategy at $t = 0$: $\boldsymbol{U}_0 = (\kappa - 1)\boldsymbol{W}_0$,
- then the information available at $t = 1$ is $\boldsymbol{Y}_1 = \kappa \boldsymbol{W}_0 + \boldsymbol{W}_1$,
- so that if κ is big enough, $\boldsymbol{Y}_1/\kappa \approx \boldsymbol{W}_0$:
- the decision maker at $t = 1$ may accurately know $\boldsymbol{W}_0$.

But increasing κ increases the first cost term $\alpha \boldsymbol{U}_0^2$...

In the classical case $\boldsymbol{Y}_0 = \boldsymbol{X}_0$, $\boldsymbol{Y}_1 = (\boldsymbol{X}_0, \boldsymbol{X}_1 + \boldsymbol{W}_1)$ (perfect memory), the optimal solution is known: $(\boldsymbol{U}_0^\sharp, \boldsymbol{U}_1^\sharp) = (0, \boldsymbol{X}_0)$.

Witsenhausen's Couterexample (2)

Let's try to give the feeling of how dual effect works.

In order to have the second cost term $\boldsymbol{X}_2^2$ as close as possible to zero, we have to guess the value of $\boldsymbol{W}_0$ at $t = 1$!
(indeed we have $\boldsymbol{X}_2 = \boldsymbol{X}_0 + \boldsymbol{U}_0 - \boldsymbol{U}_1$, with $\boldsymbol{X}_0 = \boldsymbol{W}_0$ and $\boldsymbol{U}_0 \preceq \boldsymbol{W}_0$).

- If we use a linear strategy at $t = 0$: $\boldsymbol{U}_0 = (\kappa - 1)\boldsymbol{W}_0$,
- then the information available at $t = 1$ is $\boldsymbol{Y}_1 = \kappa \boldsymbol{W}_0 + \boldsymbol{W}_1$,
- so that if κ is big enough, $\boldsymbol{Y}_1/\kappa \approx \boldsymbol{W}_0$:
- the decision maker at $t = 1$ may accurately know $\boldsymbol{W}_0$.

But increasing κ increases the first cost term $\alpha \boldsymbol{U}_0^2$...

In the classical case $\boldsymbol{Y}_0 = \boldsymbol{X}_0$, $\boldsymbol{Y}_1 = (\boldsymbol{X}_0, \boldsymbol{X}_1 + \boldsymbol{W}_1)$ (perfect memory), the optimal solution is known: $(\boldsymbol{U}_0^\sharp, \boldsymbol{U}_1^\sharp) = (0, \boldsymbol{X}_0)$.

Witsenhausen's Couterexample (2)

Let's try to give the feeling of how dual effect works.

In order to have the second cost term $\boldsymbol{X}_2^2$ as close as possible to zero, we have to guess the value of $\boldsymbol{W}_0$ at $t = 1$!
(indeed we have $\boldsymbol{X}_2 = \boldsymbol{X}_0 + \boldsymbol{U}_0 - \boldsymbol{U}_1$, with $\boldsymbol{X}_0 = \boldsymbol{W}_0$ and $\boldsymbol{U}_0 \preceq \boldsymbol{W}_0$).

- If we use a linear strategy at $t = 0$: $\boldsymbol{U}_0 = (\kappa - 1)\boldsymbol{W}_0$,
- then the information available at $t = 1$ is $\boldsymbol{Y}_1 = \kappa \boldsymbol{W}_0 + \boldsymbol{W}_1$,
- so that if κ is big enough, $\boldsymbol{Y}_1/\kappa \approx \boldsymbol{W}_0$:
- the decision maker at $t = 1$ may accurately know $\boldsymbol{W}_0$.

But increasing κ increases the first cost term $\alpha \boldsymbol{U}_0^2$...

In the classical case $\boldsymbol{Y}_0 = \boldsymbol{X}_0$, $\boldsymbol{Y}_1 = (\boldsymbol{X}_0, \boldsymbol{X}_1 + \boldsymbol{W}_1)$ (perfect memory), the optimal solution is known: $(\boldsymbol{U}_0^\sharp, \boldsymbol{U}_1^\sharp) = (0, \boldsymbol{X}_0)$.

Witsenhausen's Couterexample (2)

Let's try to give the feeling of how dual effect works.

In order to have the second cost term $\boldsymbol{X}_2^2$ as close as possible to zero, we have to guess the value of $\boldsymbol{W}_0$ at $t = 1$!
(indeed we have $\boldsymbol{X}_2 = \boldsymbol{X}_0 + \boldsymbol{U}_0 - \boldsymbol{U}_1$, with $\boldsymbol{X}_0 = \boldsymbol{W}_0$ and $\boldsymbol{U}_0 \preceq \boldsymbol{W}_0$).

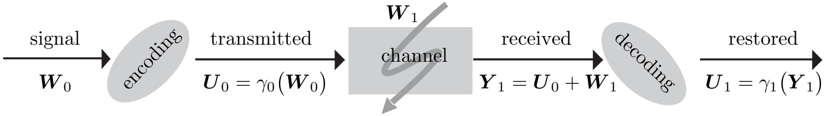
- If we use a linear strategy at $t = 0$: $\boldsymbol{U}_0 = (\kappa - 1)\boldsymbol{W}_0$,
- then the information available at $t = 1$ is $\boldsymbol{Y}_1 = \kappa \boldsymbol{W}_0 + \boldsymbol{W}_1$,
- so that if κ is big enough, $\boldsymbol{Y}_1/\kappa \approx \boldsymbol{W}_0$:
- the decision maker at $t = 1$ may accurately know $\boldsymbol{W}_0$.

But increasing κ increases the first cost term $\alpha \boldsymbol{U}_0^2 \ldots$

In the classical case $\boldsymbol{Y}_0 = \boldsymbol{X}_0$, $\boldsymbol{Y}_1 = (\boldsymbol{X}_0, \boldsymbol{X}_1 + \boldsymbol{W}_1)$ (perfect memory), the optimal solution is known: $(\boldsymbol{U}_0^\sharp, \boldsymbol{U}_1^\sharp) = (0, \boldsymbol{X}_0)$.

The Noisy Communication Channel

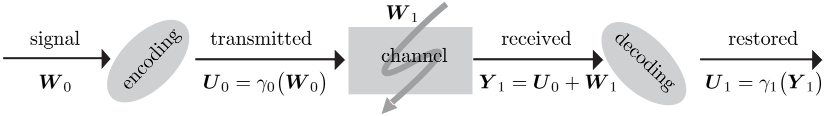
Two agents try to communicate through a noisy channel. The first agent gets a message, here simply a random variable W_0, and he wants to communicate it to the other agent. The first agent knows that the channel adds a noise W_1 to the message, so he choose to encode the original signal into another variable $U_0 = \gamma_0(W_0)$ sent through the channel. The second agent receives the noisy message $U_0 + W_1$, and make a decision $U_1 = \gamma_1(U_0 + W_1)$ about what was the original message W_0 by decoding, in an optimal manner, the received signal.



Similar to the Witsenhausen's couterexample.

The Noisy Communication Channel

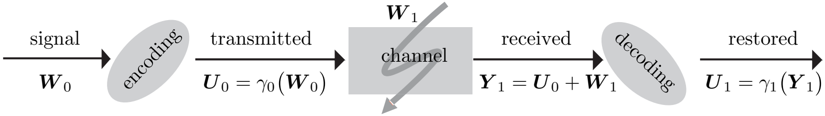
Two agents try to communicate through a noisy channel. The first agent gets a message, here simply a random variable $\boldsymbol{W}_0$, and he wants to communicate it to the other agent.The first agent knows that the channel adds a noise $\boldsymbol{W}_1$ to the message, so he choose to encode the original signal into another variable $\boldsymbol{U}_0 = \gamma_0(\boldsymbol{W}_0)$ sent through the channel. The second agent receives the noisy message $\boldsymbol{U}_0 + \boldsymbol{W}_1$, and make a decision $\boldsymbol{U}_1 = \gamma_1(\boldsymbol{U}_0 + \boldsymbol{W}_1)$ about what was the original message $\boldsymbol{W}_0$ by decoding, in an optimal manner, the received signal.



Similar to the Witsenhausen's couterexample.

The Noisy Communication Channel

Two agents try to communicate through a noisy channel. The first agent gets a message, here simply a random variable $\boldsymbol{W}_0$, and he wants to communicate it to the other agent.The first agent knows that the channel adds a noise $\boldsymbol{W}_1$ to the message, so he choose to encode the original signal into another variable $\boldsymbol{U}_0 = \gamma_0(\boldsymbol{W}_0)$ sent through the channel. The second agent receives the noisy message $\boldsymbol{U}_0 + \boldsymbol{W}_1$, and make a decision $\boldsymbol{U}_1 = \gamma_1(\boldsymbol{U}_0 + \boldsymbol{W}_1)$ about what was the original message $\boldsymbol{W}_0$ by decoding, in an optimal manner, the received signal.



Similar to the Witsenhausen's couterexample.

Goals of this Part of the Course

Objectives

- **General objective:** present numerical methods (convergence results, discretization schemes, algorithms...) in order to be able to solve optimization problems in a stochastic framework.
- **Specific objective:** be able to deal with large scale system problems for which standard methods are no more effective (dynamic programming, curse of dimensionality).

Problems under consideration

- **Open-loop problems:** decisions do not depend on specific *observation* of the uncertainties.
- **Closed-loop problems:** available observations reveal some information and decisions depend on these observations, so that it is mandatory to model the information structure.

Goals of this Part of the Course

Objectives

- **General objective:** present numerical methods (convergence results, discretization schemes, algorithms. . .) in order to be able to solve optimization problems in a stochastic framework.
- **Specific objective:** be able to deal with large scale system problems for which standard methods are no more effective (dynamic programming, curse of dimensionality).

Problems under consideration

- **Open-loop problems:** decisions do not depend on specific *observation* of the uncertainties.
- **Closed-loop problems:** available observations reveal some information and decisions depend on these observations, so that it is mandatory to model the information structure.

Goals of this Part of the Course

Objectives

- **General objective:** present numerical methods (convergence results, discretization schemes, algorithms. . .) in order to be able to solve optimization problems in a stochastic framework.
- **Specific objective:** be able to deal with large scale system problems for which standard methods are no more effective (dynamic programming, curse of dimensionality).

Problems under consideration

- **Open-loop problems:** decisions do not depend on specific *observation* of the uncertainties.
- **Closed-loop problems:** available observations reveal some information and decisions depend on these observations, so that it is mandatory to model the information structure.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (1)

Consider Problem (1) without explicit constraint Θ, and suppose that J is in fact the expectation of a function j, depending on a random variable $\boldsymbol{W}$ defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on a measurable space $(\mathbb{W}, \mathcal{W})$:

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

Then the optimization problem writes

$$\min_{u \in U^{\mathrm{ad}}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

The decision u is a deterministic variable, which only depends on the probability law of $\boldsymbol{W}$ (and not on on-line observations of $\boldsymbol{W}$). The information structure is trivial, but...

$\rightsquigarrow$ **main difficulty:** calculation of the expectation.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (1)

Consider Problem (1) without explicit constraint Θ, and suppose that J is in fact the expectation of a function j, depending on a random variable $\boldsymbol{W}$ defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on a measurable space $(\mathbb{W}, \mathcal{W})$:

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

Then the optimization problem writes

$$\min_{u \in U^{\mathrm{ad}}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

The decision u is a deterministic variable, which only depends on the probability law of $\boldsymbol{W}$ (and not on on-line observations of $\boldsymbol{W}$). The information structure is trivial, but...

$\rightsquigarrow$ **main difficulty:** calculation of the expectation.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (1)

Consider Problem (1) without explicit constraint Θ, and suppose that J is in fact the expectation of a function j, depending on a random variable $\boldsymbol{W}$ defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on a measurable space $(\mathbb{W}, \mathcal{W})$:

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) \ .$$

Then the optimization problem writes

$$\min_{u \in U^{\mathrm{ad}}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) \ .$$

The decision u is a deterministic variable, which only depends on the probability law of $\boldsymbol{W}$ (and not on on-line observations of $\boldsymbol{W}$). The information structure is trivial, but...

$\rightsquigarrow$ **main difficulty:** calculation of the expectation.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (1)

Consider Problem (1) without explicit constraint Θ, and suppose that J is in fact the expectation of a function j, depending on a random variable $\boldsymbol{W}$ defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and valued on a measurable space $(\mathbb{W}, \mathcal{W})$:

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

Then the optimization problem writes

$$\min_{u \in U^{\mathrm{ad}}} \mathbb{E}\big(j(u, \boldsymbol{W})\big) .$$

The decision u is a deterministic variable, which only depends on the probability law of $\boldsymbol{W}$ (and not on on-line observations of $\boldsymbol{W}$). The information structure is trivial, but. . .

$\rightsquigarrow$ **main difficulty:** calculation of the expectation.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (2)

Solution using Exact Quadrature

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) \quad , \quad \nabla J(u) = \mathbb{E}\big(\nabla_u j(u, \boldsymbol{W})\big) \ .$$

Projected gradient algorithm:

$$u^{(k+1)} = \mathrm{proj}_{U^{\mathrm{ad}}}\Big(u^{(k)} - \epsilon \nabla J(u^{(k)})\Big) \ .$$

Sample Average Approximation (SAA)

Obtain a realization $(w^{(1)}, \ldots, w^{(k)})$ of a k-sample of $\boldsymbol{W}$ and minimize the Monte Carlo approximation of J:

$$u^{(k)} \in \arg\min_{u \in U^{\mathrm{ad}}} \frac{1}{k} \sum_{l=1}^{k} j(u, w^{(l)}) \ .$$

Note that $u^{(k)}$ depends on the realization $(w^{(1)}, \ldots, w^{(k)})$!

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (2)

Solution using Exact Quadrature

$$J(u) = \mathbb{E}\big(j(u, \boldsymbol{W})\big) \quad , \quad \nabla J(u) = \mathbb{E}\big(\nabla_u j(u, \boldsymbol{W})\big) \ .$$

Projected gradient algorithm:

$$u^{(k+1)} = \mathrm{proj}_{U^{\mathrm{ad}}}\Big(u^{(k)} - \epsilon \nabla J(u^{(k)})\Big) \ .$$

Sample Average Approximation (SAA)

Obtain a realization $(w^{(1)}, \ldots, w^{(k)})$ of a k-sample of $\boldsymbol{W}$ and minimize the Monte Carlo approximation of J:

$$u^{(k)} \in \arg\min_{u \in U^{\mathrm{ad}}} \frac{1}{k} \sum_{l=1}^{k} j(u, w^{(l)}) \ .$$

Note that $u^{(k)}$ depends on the realization $(w^{(1)}, \ldots, w^{(k)})$!

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (3)

Stochastic Gradient Method

Underlying ideas:

- use an easily computable approximation of ∇J based on realizations $(w^{(1)}, \ldots, w^{(k)}, \ldots)$ of samples of $\boldsymbol{W}$,
- incorporate the realizations one by one into the algorithm.

These considerations lead to the following algorithm:

$$u^{(k+1)} = \mathrm{proj}_{U^{\mathrm{ad}}} \left(u^{(k)} - \epsilon^{(k)} \nabla_u j(u^{(k)}, w^{(k+1)}) \right) .$$

Iterations of the gradient algorithm are used a) to move towards the solution *and* b) to refine the Monte-Carlo sampling process.

$\rightsquigarrow$ Topic of the first three lessons.

Extension of Problem $(\mathcal{P}_G)$ — Open-Loop Case (3)

Stochastic Gradient Method

Underlying ideas:

- use an easily computable approximation of ∇J based on realizations $(w^{(1)}, \ldots, w^{(k)}, \ldots)$ of samples of $\boldsymbol{W}$,
- incorporate the realizations one by one into the algorithm.

These considerations lead to the following algorithm:

$$u^{(k+1)} = \mathrm{proj}_{U^{\mathrm{ad}}} \left(u^{(k)} - \epsilon^{(k)} \nabla_u j(u^{(k)}, w^{(k+1)}) \right) .$$

Iterations of the gradient algorithm are used a) to move towards the solution *and* b) to refine the Monte-Carlo sampling process.

$\rightsquigarrow$ **Topic of the first three lessons.**

Open-loop stochastic optimization problems

- Stochastic gradient method overview
  - Stochastic gradient algorithm and stochastic approximation.
  - Asymptotic efficiency and averaging.
  - Practical considerations.
  - Machine Learning point of view
- Generalized stochastic gradient method
  - Auxiliary Problem Principle in the deterministic setting.
  - Auxiliary Problem Principle in the stochastic setting.
  - Extension to constrained problems.
- Applications of the stochastic gradient method
  - Simple exercices.
  - Option pricing problem and variance reduction.
  - Spatial rendez-vous under probability constraint.

Extension of Problem $(\mathcal{P}_G)$ — Closed-Loop Case

Algebraic approach: Stochastic Programming

Rather than looking for the solution of the problem as feedback functions depending on information (Dynamic Programming point of view), we seek at obtaining the problem solution as random variables satisfying the information constraints:

$$\sigma(\boldsymbol{U}) \subset \sigma(\boldsymbol{Y}) \ .$$

- First issue: characterize the class of problems that can be solved by this approach. The problem is much more intricate if dual effect is present ($\boldsymbol{Y}$ depends on $\boldsymbol{U}$).
- Second issue: obtain a finite approximation of the problem, and more specifically discretize the information constraints.

⇝ **Topic of the penultimate lesson.**

Extension of Problem $(\mathcal{P}_G)$ — Closed-Loop Case

Algebraic approach: Stochastic Programming

Rather than looking for the solution of the problem as feedback functions depending on information (Dynamic Programming point of view), we seek at obtaining the problem solution as random variables satisfying the information constraints:

$$\sigma(\boldsymbol{U}) \subset \sigma(\boldsymbol{Y}) \; .$$

- First issue: characterize the class of problems that can be solved by this approach. The problem is much more intricate if dual effect is present ($\boldsymbol{Y}$ depends on $\boldsymbol{U}$).
- Second issue: obtain a finite approximation of the problem, and more specifically discretize the information constraints.

$\rightsquigarrow$ Topic of the penultimate lesson.

Extension of Problem $(\mathcal{P}_G)$ — Closed-Loop Case

Algebraic approach: Stochastic Programming

Rather than looking for the solution of the problem as feedback functions depending on information (Dynamic Programming point of view), we seek at obtaining the problem solution as random variables satisfying the information constraints:

$$\sigma(\boldsymbol{U}) \subset \sigma(\boldsymbol{Y}) \ .$$

- First issue: characterize the class of problems that can be solved by this approach. The problem is much more intricate if dual effect is present ($\boldsymbol{Y}$ depends on $\boldsymbol{U}$).
- Second issue: obtain a finite approximation of the problem, and more specifically discretize the information constraints.

$\rightsquigarrow$ **Topic of the penultimate lesson.**

Extension of the Dynamic Problem

Dynamic Programming and decomposition

On the one hand, Dynamic Programming can not be used in a straightforward manner to large scale stochastic optimal control problems. On the other hand, decomposition and coordination methods such as Lagrangian relaxation apply, but subproblems can not be solved optimally by DP.

- First issue: have a close look to stochastic optimal control problems in discrete time in order to highlight the associated opportunities of decomposition.
- Second issue: devise an approximate decomposition and coordination method such that subproblems can be solved by Dynamic Programming.

$\rightsquigarrow$ **Topic of the last lesson.**

Extension of the Dynamic Problem

Dynamic Programming and decomposition

On the one hand, Dynamic Programming can not be used in a straightforward manner to large scale stochastic optimal control problems. On the other hand, decomposition and coordination methods such as Lagrangian relaxation apply, but subproblems can not be solved optimally by DP.

- First issue: have a close look to stochastic optimal control problems in discrete time in order to highlight the associated opportunities of decomposition.
- Second issue: devise an approximate decomposition and coordination method such that subproblems can be solved by Dynamic Programming.

$\rightsquigarrow$ Topic of the last lesson.

Extension of the Dynamic Problem

Dynamic Programming and decomposition

On the one hand, Dynamic Programming can not be used in a straightforward manner to large scale stochastic optimal control problems. On the other hand, decomposition and coordination methods such as Lagrangian relaxation apply, but subproblems can not be solved optimally by DP.

- First issue: have a close look to stochastic optimal control problems in discrete time in order to highlight the associated opportunities of decomposition.
- Second issue: devise an approximate decomposition and coordination method such that subproblems can be solved by Dynamic Programming.

$\rightsquigarrow$ **Topic of the last lesson.**

Closed-loop stochastic optimization problems

- Stochastic optimization and discretization
  - Stochastic Programming: the scenario tree method.
  - Stochastic Optimal Control and discretization puzzles.
  - General convergence result.
- Stochastic optimization and decomposition
  - Decomposition and coordination.
  - Dual Approximate Dynamic Programming.
  - Theoretical questions.

1. General Introduction
   - Decision Making as an Optimization Problem
   - Facing the Uncertainty
   - The Role of Information

2. Problem Formulation and Information Structure
   - The Stochastic Programming Approach
   - The Stochastic Optimal Control Approach
   - Examples

3. Content of the course
   - Part of the Course by Pierre Carpentier
   - Part of the Course by Vincent Leclère

Mathematical foundations of stochastic optimization

- Convex analysis and convex optimization
    - Fenchel conjugate, subdifferential calculus.
    - Lagrangian duality and duality by perturbations.
    - Marginal interpretation of multipliers.
- Integration and measure theory
    - Subdifferential of an expectation, normal integrands.
    - Exchange of min and expectation.
    - Uniform law of large numbers.
    - Newsvendor problem
- Stochastic programming and the two-stage case
    - Optimization under uncertainty.
    - Stochastic programming approach.
    - Information and discretization.

Dynamic stochastic optimization

- Scenario decomposition: L-Shaped and Progressive Hedging
  - Information frameworks.
  - Lagrangian decomposition.
  - L-Shaped decomposition method.
- Bellman operators and Stochastic Dynamic Programming
  - Bellman operators abstract framework.
  - Stochastic Dynamic Programming.
- Stochastic Dual Dynamic Programming (SDDP)
  - Kelley's algorithm.
  - Deterministic case.
  - Stochastic case.