Technical Report

DECENTRALIZED STOCHASTIC CONTROL

Submitted in partial fulfillment of the requirements for the course of

ENPM667 – CONTROL OF ROBOTIC SYSTEMS

by

ADHEESH CHATTERJEE
UID – 116236935
M.ENG ROBOTICS

PROJECT SUPERVISOR
Prof Waseem Mallik

UNIVERSITY OF MARYLAND 1856

A. JAMES CLARK
SCHOOL OF ENGINEERING

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	NOTATION	i
	ABSTRACT	ii
	SCOPE OF PAPER	iii
	SCOPE OF REPORT	iii
1	INTRODUCTION	
	1.1 STOCHASTIC CONTROL	1
	1.2 BAYESIAN PROBABILITY	2
	1.3 CENTRALIZED STOCHASTIC CONTROL	3
	1.4 DECENTRALIZED STOCHASTIC CONTROL	3
	1.5 MARKOV MODEL	3
	1.5.1 MARKOV DECISION PROCESS	4
	1.5.2 PARTIALLY OBSERVABLE MARKOV DECISION PROCESS	4
	1.5.3 MARKOV CHAIN	4
	1.5.4 HIDDEN MARKOV MODEL	4
	1.6 DISCRETE CONTROL: BELLMAN EQUATIONS	5
	1.7 DYNAMIC PROGRAMMING	5
	1.8 VALUE ITERATION AND POLICY ITERATION	6
	1.9 MARKOV DECISION PROCESS	7
	1.10 CONTINUOUS CONTROL: HAMILTON-JACOBI-JACOBI EQUATIONS	8
	1.11 DETERMINISTIC CONTROL: PONTRYAGIN'S MAXIMUM PRINCIPLE	10
	1.12 SIGMA FIELD	11
	1.13 CANONICAL FILTRATION	11
	1.14 COOPERATIVE GAME THEORY	11
	1.15 DYNAMIC GAME THEORY	11
	1.16 NASH EQUILIBRIA	11
	1.17 ITO CALCULUS	12
	1.18 FINITE VS INFINITE HORIZON	12
	1.19 MULTIVARIATE GAUSSIAN DISTRIBUTION	12
	1.20 REALIZATION	13
2	MODEL & PROBLEM FORMULATION	14
	2.1 STATE, OBSERVATION AND CONTROL PROCESSES	14
	2.2 INFORMATION STRUCTURE	14
	2.3 CONTROL STRATEGIES AND PROBLEM FORMULATION	15
	2.4 RELATIONSHIP TO OTHER MODELS	15
	2.5 CONCEPTUAL DIFFICULTIES IN FINDING OPTIMAL SOLUTION	16

3 OVERVIEW OF CENTRALIZED STOCHASTIC CONTROL 17

4 CONCEPTUAL DIFFICULTIES IN DYNAMIC PROGRAMMING FOR DECENTRALIZED STOCHASTIC CONTROL 19

5 THE PERSON-BY-PERSON APPROACH 20

5.1 STRUCTURE OF OPTIMAL CONTROL STRATEGIES 20

5.2 COUPLED DYNAMIC PROGRAM FOR PERSON-BY-PERSON OPTIMAL SOLUTION 20

6 THE COMMON-INFORMATION APPROACH 22

6.1 BASIC MODEL 22

6.2 EQUIVALENT CENTRALIZED STOCHASTIC CONTROL MODEL 25

7 COMPARISON BETWEEN COMMON INFORMATION APPROACH AND PERSON-BY-PERSON APPROACH 30

8 CONCLUSION 31

9 APPENDICES 33

9.1 APPENDIX A (DEDINITIONS) 33

9.2 APPENDIX B (THEOREMS) 35

9.3 APPENDIX C (PROOFS) 38

9.4 APPENDIX D (PROPOSITIONS) 40

10 BIBLIOGRAPHY 41

11 ACKNOWLEDGEMENTS 45

LIST OF NOTATIONS

Notation Used	Definition
Upper Case Letters	Random variables
Lower Case Letters	Realizations
Calligraphic Letters	Space of Realization
Subscripts	Time Index
Superscripts	Index Controllers
For integers $a \leq b$, $X_{a:b}$	Set $\{X_a, X_{a+1} \dots, X_b\}$
For integers $a \geq b$, $X_{a:b}$	Empty set
$P(\cdot)$	Probability of Event
$E[\cdot]$	Expectation of Random Variable
For function *g*, $P^g(\cdot)$	Probability measure depending on the choice of function *g*
For function *g*, $E^g[\cdot]$	Expectation depending on the choice of function *g*
$Z_{>0}$	Set of Positive Integers
$\mathbb{R}$	Set of Real Numbers
$\prod_b^a(\cdot)$	Product from a to b
Δ	Delta Operator (Δf$_k$ = f$_{k+1}$−f$_k$)
Inf	Infimum
Sup	Supremum

ABSTRACT

Stochastic control is a subfield of control theory that deals with the existence of uncertainty either in observations or in the noise that drives the evolution of the system. Decentralized stochastic control refers to the multi-stage optimization of the dynamical system by multiple controllers which have access to different information. However, decentralization of this information gives rise to new conceptual challenges that require new solution approaches. In this expository paper, the notion of an information-state to explain the two commonly used solution approaches to decentralized control: the person-by-person approach and the common-information approach are discussed.

Keywords: *Decentralized stochastic control; Dynamic programming; Team theory; Information structures*

SCOPE OF PAPER

- The focus of this expository paper is to highlight conceptual challenges of decentralized control and explain the intuition behind the solution approaches.
- No new results are presented in this paper; rather new insights and connections between existing results is presented.
- Since the paper focuses solely on conceptual understanding, no proofs are presented and the technical details, in particular, measurability concerns, are ignored in the description.
- Initially the paper formulates a general model, its information structure and control strategies before comparing the relationship to other models and presenting the conceptual difficulty in finding the optimal solution. Finally an example is presented to further explain the results
- An overview of centralized stochastic control is presented along with theorems for information-state processes. Finally an example is presented to illustrate the concepts presented
- The conceptual difficulties in dynamic programming for decentralized stochastic control are explained at depth
- The paper then presents two commonly used solution approaches to decentralized control – the person-by-person approach and the common-information approach. An example of both approaches are presented to further illustrate the concepts

SCOPE OF REPORT

This report provides a technical summary of the work done by Aditya Mahajan and Mehnaz Menon in the paper “Decentralized Stochastic Control”.
The paper was supported by Fonds de recherche du Québec–Nature et technologies (FRQ-NT), Establishment of New Researcher Grant 166065 and by Natural Science and Engineering Research Council of Canada (NSERC) Discovery Grant 402753-11. The paper was published online in the journal *Annals of Operations Research June 2016, Volume 241, Issue 1–2, pp 109–126* by Springer US.

This report analyzes all the results presented in this paper and provides an in-depth analysis of the proofs of all the models and concepts presented. The report also looks at the basic concepts mentioned in the paper that are part of optimal stochastic control though not specifically explained in the paper. This report does not, however, explain the examples as they require a lot of knowledge in the field of controls to be easily interpretable.

CHAPTER 1
INTRODUCTION

There exists a drastic difference between a completely autonomous humanoid robot and a programmable industrial robot. In response to this, two paths emerged on how to develop the perfect robot system. One approach which was highly intellectual was to equip the robot with sensors so that it could interface successfully with the uncertain world. Another approach which was quite expensive tailored the robot's corner of the world so that order and predictability would reign.

But a robot with sensing and control exactly equal to that of a human has not yet been built, and it is equally impossible to remove all uncertainty from the robot's environment. Thus, a compromise between the extremes emerged, and as robot sensing advanced, the robot's tolerance of uncertainty advanced. Thus, stochastic control was developed for the purpose of robot sensing.

Stochastic control, and the associated principle of dynamic programming, have roots in statistical sequential analysis (Arrow et al. 1949) and have been used in various application domains including operations research (Powell 2007), economics (Stokey and Lucas Robert 1989), engineering (Bertsekas 1995), computer science (Russell and Norvig 1995), and mathematics (Bellman 1957).

Decentralized stochastic control started with seminal work of Radner (1962), Marschak and Radner (1972) on static systems that arise in organizations and of Witsenhausen (1971 and 1973) on dynamic systems that arise in systems and control. Decentralized stochastic control is fundamentally different from, and significantly more challenging than, centralized stochastic control.

Centralized Stochastic Control problems can primarily be solved by dynamic programming, but DP does not directly work in decentralized stochastic control. New ways of thinking need to be developed to address information decentralization. In this paper, the conceptual challenges of decentralized control are highlighted and the intuition behind the solution approaches are explained.

1.1 Stochastic Control

Stochastic control or stochastic optimal control is a subfield of control theory that deals with the evolution of the system in the presence of noise and existence of uncertainty in observations. The framework architecture assumes that random noise with known probability distribution affects the evolution and observation of the state variables. This is done in a Bayesian probability-driven fashion. The aim of stochastic control is to design the time path of the controlled variables that perform the desired control task with minimum cost despite the presence of this noise. It may be either discrete time or continuous time.

1.2 Bayesian Probability

Bayesian probability is a concept of probability, in which, instead of *frequency*, probability is interpreted as reasonable expectation representing a state of knowledge. Around 200 years ago, the foundations of Bayesian probability theory were laid down by people such as Bernoulli, Bayes, and Laplace. However, since its inception, it has been held suspect or controversial by modern statisticians but the last few decades though have seen the occurrence of a "Bayesian revolution", and Bayesian probability theory is now commonly employed in many scientific disciplines, from astrophysics to neuroscience. It is most often used to judge the relative validity of hypotheses in the face of noisy, sparse, or uncertain data, or to adjust the parameters of a specific model.

To further illustrate the concept, a classic example is explained. Here Bayesian probability theory is employed in the problem of estimation. We must guess the value of an underlying parameter from an observation that is corrupted by noise.

Let's say we have a quantity x and the observation of this quantity y is corrupted by additive Gaussian noise n with zero mean –

$$y = x + n$$

So, we have to make the best guess possible of the value of x, given the observation y. Considering we know the probability distribution of x given y i.e. $P(x|y)$, then, by definition, we need to find x that maximizes –

$$\hat{x} = \arg\max_x P(x|y)$$

We can also achieve the same outcome by minimizing the mean squared error of our guesses of x. Then we pick the mean of $P(x|y)$ and hence –

$$\hat{x} = \int x\, P(x|y)\, dx$$

So essentially, we only need to know $P(x|y)$ to make an optimal guess. We can do this using Bayes' Rule as –

$$P(x|y) = \frac{P(y|x)\, P(x)}{P(y)}$$

Now, we need to only figure out $P(y|x)$ and $P(x)$ as y is the observation.So $P(y|x)$ can be rewritten as $P(n + x|x)$ as $y = n + x$ and hence –

$$P(y|x) = P(n + x|x) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(y-x)^2}{2\sigma_n^2}}$$

where σ_n^2 is the variance of the noise.

For $P(x)$, we need to have prior knowledge of x and its mean $\bar{x}$. Let us assume the variance is 1 and hence we have the Gaussian Distribution –

$$P(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\bar{x})^2}{2}}$$

Hence, we can write –

$$P(x|y) \propto P(y|x)\, P(x)$$

$$= e^{-\frac{(y-x)^2}{2\sigma_n^2}}\, e^{-\frac{(x-\bar{x})^2}{2}}$$

$$= e^{-\frac{1}{2}\left[\frac{(y-x)^2}{\sigma_n^2} + (x-\bar{x})^2\right]}$$

Thus the x which maximizes $P(x|y)$ is the same as the one that maximizes the exponent brackets which may be found by simple algebra –

$$\hat{x} = \frac{y + \bar{x}\sigma_n^2}{1 + \sigma_n^2}$$

Bayesian Probability is very vast field and has concepts of Generative Models, Inference (Perception), Learning (Adaptation) and Neural Decoding that have great significance in stochastic control but are beyond the scope of this report.

1.3 Centralized Stochastic Control

The fundamental assumption of centralized stochastic control is that the decision at each stage are made by a single controller that has perfect recall, i.e. the controller remembers its past observations and decisions. Dynamic Programming is the primary solution concept of centralized stochastic control

1.4 Decentralized Stochastic Control

The fundamental assumption in centralized stochastic control is violated in many modern applications where decisions are made by multiple controllers that have access to different information. The multi-stage optimization of such systems is called *decentralized stochastic control* or *dynamic team theory*. Dynamic Programming does not directly work in decentralized stochastic control and different approaches need to be taken to address information decentralization.

1.5 Markov Models

A Markov Model is a stochastic model used to model randomly changing systems. Here we assume that future states depend only on the current state and not on the events that occur before it. There are four types of Markov models which are all used in different situations depending on whether every sequential state is observable or not, and whether the system is to be adjusted based on observations made.

MARKOV MODELS		State Transitions Are Controllable	
		Yes (System is controlled)	No (System is autonomous)
System State Is Fully Observable	Yes	Markov Decision Process (MDP)	Markov Chain
	No	Partially Observable Markov Decision Process (POMDP)	Hidden Markov Model (HMM)

1.5.1 Markov Decision Process (MDP)

A Markov Decision Process (MDP) is a discrete time stochastic control process. It provides a mathematical framework for modelling decision making in situations where outcomes are partly random and partly under the control of a decision maker. MDPs are useful for studying optimization problems solved via dynamic programming and reinforcement learning.

1.5.2 Partially Observable Markov Decision Process (POMDP)

A Partially Observable Markov Decision Process (POMDP) is a generalization of a Markov decision process (MDP). A POMDP models an agent decision process in which it is assumed that the system dynamics are determined by an MDP, but the agent cannot directly observe the underlying state. Instead, it must maintain a probability distribution over the set of possible states, based on a set of observations and observation probabilities, and the underlying MDP.

1.5.3 Markov Chain

The simplest Markov model is the Markov chain. It models the state of a system with a random variable that changes with respect to time. By the Markov property, we can say that the distribution for this variable depends only on the distribution of previous state.

1.5.4 Hidden Markov Model (HMM)

A Hidden Markov model is a Markov chain for which the state is only partially observable. In other words, observations are related to the state of the system, but they are typically insufficient to precisely determine the state. Several well-known algorithms for solving hidden Markov models exist like the *Viterbi algorithm* which computes the most-likely corresponding sequence of states, the *Forward algorithm* which computes the probability of the sequence of observations, and the *Baum–Welch algorithm* which estimates the starting probabilities, the transition function, and the observation function of a hidden Markov model. The discussions of these algorithms is beyond the scope of this report

1.6 Discrete Control – Bellman Equations

Bellman equations, named after Richard E. Bellman, are a necessary condition for optimality associated with the mathematical optimization method of dynamic programming.

Let $x \in \mathcal{X}$ denote the state of an agent's environment and let $u \in \mathcal{U}(x)$ be the action (or control) which the agent chooses while at state x. Let's assume both $\mathcal{X}$ and $\mathcal{U}$ are finite sets. Let $next(x, u) \in \mathcal{X}$ denote the state which results from applying action u in state x, and $cost(x, u) \geq 0$ be the cost of applying action u in state x.

To further illustrate the concept, let's take an example, x is the city where we are now, u is the flight we want to take. Let $next(x, u)$ be the city where that flight lands, and $cost(x, u)$ be the price of the ticket. By definition, we now have an optimal control problem that will help us find the cheapest way to fly to our destination.

This problem can be formalized as follows: Find an action sequence $(u_0 u_1, \dots u_{n-1})$ and corresponding state sequence $(x_0 x_1, \dots x_n)$ minimizing the total cost

$$J(x, u) = \sum_{k=0}^{n-1} cost(x_k, u_k)$$

where $x_{k+1} = next(x, u)$ and $u_k \in \mathcal{U}(u_k)$. The initial state $x_0 = x^{init}$ and destination $x_n = x^{dest}$ are given. We can also say that if $cost(x, u) = 1$ for all (x, u) the problem reduces to finding the shortest path from x^{init} to x^{dest} which can easily be solved by concepts of discrete mathematics.

The Bellman Equations for the problem can be found by Dynamic Programming.

Principle of Optimality: An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

1.7 Dynamic Programming

Dynamic programming refers to simplifying a decision by breaking it down into a sequence of decision steps over time. This is done by defining a sequence of value functions $V_1, V_2 \dots V_n$ taking y as an argument representing the state of the system at time $i, i \in \{1, \dots, n\}$. The definition of $V_n(y)$ is the value obtained in state y at the last time n. The values V_i at earlier times $i = n - 1, n - 2 \dots, 2, 1$ can be found by working backwards, using the Bellman equation.

For $i = 2, \dots, n$, V_{i-1} at any state y is calculated from V_i by maximizing a simple function of the gain from a decision at time $i - 1$ and the function V_i at the new state of the system if this decision is made. Since V_i has already been calculated for the needed states, the above operation yields V_{i-1} for those states. Finally, V_1 at the initial state of the system is the value of the optimal solution. The optimal values of the decision variables can be recovered, one by one, by tracking back the calculations already performed.

This concept bears a very close resemblance to the Markov property of stochastic processes. So, essentially, we can write the optimality principle as – the choice of optimal actions in the future is independent of the past actions which led to the present state. Thus, optimal state-action sequences can be constructed by starting at the final state and extending backwards.

This yields an optimal action $u = \pi(x) \in \mathcal{U}(x)$ for every state x. A mapping from states to actions is called control law or control policy. Once we have a control law $\pi : \mathcal{X} \to \mathcal{U}(\mathcal{X})$ we can start at any state x_0, generate action $u_0 = \pi(x_0)$, transition to state $x_1 = next(x_0, u_0)$, generate action $u_1 = \pi(x_1)$, and keep going until we reach x^{dest}.
The optimal control law π satisifies

$$\pi(x) = \arg min_{u \in \mathcal{U}(x)} \{cost(x,u) + v(next(x,u)\ )\} \tag{1}$$

The minimum above may be achieved for multiple actions in the set $\mathcal{U}(x)$, which is why π may not be unique. However, the optimal control value function v is always uniquely defined and satisfies

$$v(x) = min_{u \in \mathcal{U}(x)} \{cost(x,u) + v\big(next(x,u)\big)\} \tag{2}$$

These 2 equations are the ***Bellman equations***.

1.8 Value Iteration and Policy Iteration

For cyclic graphs although the Bellman equations are valid, we cannot apply them in a single pass. This is because the presence of cycles makes it impossible to visit each state only after all its successors have been visited. Instead the Bellman equations are treated as consistency conditions and used to design iterative relaxation schemes, much like partial differential equations (PDEs) are treated as consistency conditions and solved with corresponding relaxation schemes.

Relaxation Scheme means guessing the solution, and iteratively improving the guess so as to make it more compatible with the consistency condition.

The two main relaxation schemes are **Value Iteration** and **Policy Iteration**.

Value Iteration uses only (2).

We start with a guess $v^{(0)}$ of the optimal value function, and construct a sequence of improved guesses:

$$v^{(i+1)}(x) = min_{u \in \mathcal{U}(x)} \{cost(x,u) + v^{(i)}(next(x,u))\} \tag{3}$$

This process is guaranteed to converge to the optimal value function v in a finite number of iterations. The proof relies on the important idea of contraction mappings: one defines the approximation error $e\left(v^{(i)}\right) = max_x \left| v^{(i)}(x) - v(x) \right|$, and shows that iteration (3) causes $e\left(v^{(i)}\right)$ to decrease as i increases. In other words, the mapping $v^{(i)} \rightarrow v^{(i+1)}$ given by (3) contracts the size of $v^{(i)}$ as measured by the error norm $e\left(v^{(i)}\right)$.

Policy iteration uses both (1) and (2).

We start by guessing $\pi^{(0)}$ of the optimal control law and construct a sequence of improved guesses.

$$v^{\pi^{(i)}}(x) = cost\left(x, \pi^{(i)}(x)\right) + v^{\pi^{(i)}}\left(next\left(x, \pi^{(i)}(x)\right)\right) \tag{4}$$

$$\pi^{(i+1)}(x) = \arg min_{u \in \mathcal{U}(x)} \left\{cost(x,u) + v^{\pi^{(i)}}(next(x,u)\ )\right\} \tag{5}$$

(4) requires a separate relaxation to compute the value function $v^{\pi^{(i)}}$ for the control law $\pi^{(i)}$. This function is defined as the total cost for starting at state x and acting according to $\pi^{(i)}$ thereafter. Policy iteration can also be proven to converge in a finite number of iterations.

We cannot say algorithm is better, because each of the two nested relaxations in policy iteration converges faster than the single relaxation in value iteration. In practice both algorithms can be used depending on the problem at hand.

1.9 Markov Decision Process

Dynamic programming easily generalizes to the stochastic case where we have a probability distribution over possible next states:

$P(y|x,u)$ is the probability that the $next(x,u)$ is y

Now since it is a probability distribution, we know

$$\sum\nolimits_{y \in \mathcal{X}} P(y|x,u) = 1$$

And $$P(y|x,u) \geq 0$$

In the stochastic case the value function or (2) becomes

$$v(x) = min_{u \in \mathcal{U}(x)}\{cost(x,u) + E\left[v\left(next(x,u)\right)\right]\} \tag{6}$$

Where E is the expectation over $next(x,u)$ and is computed as

$$E\left[v\left(next(x,u)\right)\right] = \sum\nolimits_{y \in \mathcal{X}} P(y|x,u)\, v(y)$$

Equations (1),(3),(4) and (5) can be shown to generalize to the stochastic case in the same way as equation (2) does as shown above.

An optimal control problem with discrete states and actions and probabilistic state transitions is called a Markov decision process (MDP).

1.10 Continuous Control: Hamilton-Jacobi-Bellman Equation

In optimal control problems where the state $x \in \mathbb{R}^{n_x}$ and control $u \in \mathcal{U}(x) \subseteq \mathbb{R}^{n_u}$ are real valued vectors

Let us consider the stochastic equation

$$dx = f(x,u)dt + F(x,u)dw \qquad (7)$$

Or $$x(t) = x(0) + \int_0^t f\big(x(s),u(s)\big)ds + \int_0^t F\big(x(s),u(s)\big)dw(s)$$

The last term is an *ito-integral*, defined as

$$\int_0^t g(s)dw(s) = \lim_{n\to\infty} \sum_{k=0}^{n-1} g(s_k)(w(\,s_{k+1}) - w(s_k))$$

where $0 = s_0 < s_2 < \cdots < s_n = t$

Now on discretizing the time axis and obtaining results for the continuous-time case in the limit of infinitely-small time steps.

The Euler Discretization of (6) is

$$x_{k+1} = x_k + \Delta f(x_k,u_k) + \sqrt{\Delta}F(x_k,u_k)s_k$$

where Δ is the time step $\varepsilon_k \sim \mathcal{N}(0, I^{n_w})$ and $x_k = x(k\Delta)$. The $\sqrt{\Delta}$ term occurs because of the variance of the Brownian motion grows linearly with time and thus the standard deviation of the discrete time noise will scale as $\sqrt{\Delta}$. Concepts of Ito calculus are looked into in 1.17.

We now define a cost function. In finite-horizon problems, i.e. when a final time t_f is specified, it is natural to separate the total cost into a time-integral of a *cost rate* $l(x,u,t) \geq 0$, and a final cost $h(x) \geq 0$ which is only evaluated at the final state x(t_f). Thus, the total cost for a given state-control trajectory $\{x(t), u(t) \mid 0 \leq t \leq t_f\}$ is defined as

$$J\big(x(\cdot),u(\cdot)\big) = h\left(x\left(t_f\right)\right) + \int_0^{t_f} l(x(t),u(t),t)dt$$

Since we are dealing with a stochastic system, we need to find a control law $u = \pi(x,t)$ which minimizes the expected total cost for a starting given (x,t) and acts according to π afterward.

In discrete time, the total cost becomes

$$J(x,u) = h(x_n) + \Delta \sum_{k=0}^{n-1} l(x_k, u_k, k\Delta)$$

where $n = t_f/\Delta$ is the number of time steps.

We now apply dynamic programming to the time-discretized stochastic problem. The development is similar to the Markov Decision Process (1.9) case except that the state space is now infinite: it consists of $n+1$ copies of $\mathbb{R}^{n_x}$.

The state transitions are now stochastic: the probability distribution of x_{k+1} given x_k, u_k is the multivariate Gaussian

$$x_{k+1} \sim \mathcal{N}(x_k + \Delta f(x_k, u_k), \Delta S(x_k, u_k))$$

where $$S(x,u) = F(x,u)\, F(x,u)^T$$

The bellman equation for the optimal value function v is similar to (6), except now v is a function of space and time. We have

$$v(x,k) = min_{u \in \mathcal{U}(x)}\{\Delta l(x,u,k\Delta) + E[v(x + \Delta f(x,u) + \xi, k+1)]\} \tag{8}$$

where $\xi \sim \mathcal{N}\big(0, \Delta S(x,u)\big)$ and $v(x,n) = h(x)$

Consider the second order Taylor-series expansion of v, with the time index $k+1$ suppressed for clarity:

$$v(x+\delta) = v(x) + \delta^T v_x(x) + \frac{1}{2}\delta^T v_{xx}(x)\delta + \cdots$$

where $\delta = \Delta f(x,u) + \xi, v_x = \frac{\partial}{\partial x} v, v_{xx} = \frac{\partial^2}{\partial x^2} v$

Now we compute the expectation of the optimal value function at the next state using the above Taylor-series expansion and only keeping terms up to first-order in Δ. The result is:

$$E[v] = v(x) + \Delta f(x,u)^T v_x(x) + \frac{1}{2} tr(\Delta S(x,u) v_{xx}(x))$$

The trace term appears here because

$$E[\xi^T v_{xx} \xi] = E[tr(\xi\xi^T v_{xx})] = tr(Cov[\xi] v_{xx}) = tr(\Delta S v_{xx})$$

We use Ito's Lemma here which states that if x(t) is an ito diffusion with coefficient σ, then

$$dg\big(x(t)\big) = g_x\big(x(t)\big)dx(t) + \frac{1}{2}\sigma^2 g_{xx}\big(x(t)\big)dt$$

Hence, we have the v_{xx} term in $E[v]$

We now substitute this $E[v]$ in (8), move the term $v(x)$ outside the min operator (as it doesn't depend on u) and divide by Δ. Suppressing the x, u, k on the right-hand side for simplicity we have

$$\frac{v(x,k) - v(x,k+1)}{\Delta} = min_u\{l + f^T v_x + \frac{1}{2} tr(Sv_{xx})$$

We know that $t = k\Delta$ and considering the optimal value function $v(x,t)$ defined in continuous time, the LHS of the above equation becomes

$$\frac{v(x,t) - v(x,t+\Delta)}{\Delta}$$

In the limit $\Delta \rightarrow 0$, it is of a general limit form. Hence the value becomes $-\frac{\partial}{\partial t} v$, which is denoted as $-v_t$. Thus for $0 \leq t \leq t_f$ and $v\left(x, t_f\right) = h(x)$, we get

$$-v_t = min_{u \in \mathcal{U}(x)} \left\{ l(x,u,t) + f(x,u)^T v_x(x,t) + \frac{1}{2} tr\left(S(x,u) v_{xx}(x,t)\right)\right\} \qquad (9)$$

Similarly, as in the discrete case, we get the optimal control law $\pi(x,t)$ is a value of u which achieves the minimum of (9)

$$\pi(x,t) = arg\ min_{u \in \mathcal{U}(x)} \left\{ l(x,u,t) + f(x,u)^T v_x(x,t) + \frac{1}{2} tr\left(S(x,u) v_{xx}(x,t)\right)\right\} \qquad (10)$$

Equations (9) and (10) are the Hamilton-Jacobi-Bellman Equations

1.11 Deterministic Control – Pontryagin's Maximum Principle

Optimal control theory is based on two fundamental ideas –

One is dynamic programming and the associated optimality principle which was introduced by Bellman in the United States.

The other is the maximum principle, introduced by Pontryagin in the Soviet Union which applies only to deterministic problems and yields the same solutions as dynamic programming. However, unlike dynamic programming, the maximum principle avoids the problem of dimensionality.

The Maximum Principle can be derived indirectly via the HJB equation or directly via Lagrange multipliers. However, for this report, the derivation of the Maximum Principle is beyond the scope.

1.12 Sigma Field

The definition of a sigma-field requires that we have a sample space *S* along with a collection of subsets of *S*. This collection of subsets is a sigma-field if the following conditions are met:

- If the subset *A* is in the sigma-field, then so is its complement A^C.
- If A_n are countably infinitely many subsets from the sigma-field, then both the intersection and union of all of these sets are also in the sigma-field.

Further analysis of this concept is beyond the scope of this report.

1.13 Canonical Filtration

The canonical filtration is the smallest filtration to which $\{X_t, t \geq 0\}$ is adapted to $\{\mathcal{F}_t, t \geq 0\}$ if for all $t \geq 0$, the random variable X_t is measurable with respect to the σ-field $\mathcal{F}_t$. The canonical filtration of a continuous $\{X_{t,} t \geq 0\}$,is the filtration $\{\mathcal{F}_t^X, t \geq 0\}$ where $\mathcal{F}_t^X$ = $\sigma(X_u: 0 \leq u \leq t)$ where σ is a sigma field.

Further analysis of this concept is beyond the scope of this report.

1.14 Cooperative Game Theory

A cooperative game, according to the concept of game theory, is a game with competition between groups of players due to the possibility of external enforcement of cooperative behaviour. This is the opposite of non-cooperative games in which there is either no possibility to forge alliances or all agreements need to be self-enforcing.

Cooperative games are often analysed through the framework of cooperative game theory, which focuses on predicting which coalitions will form, the joint actions that groups take and the resulting collective payoffs. It is opposed to the traditional non-cooperative game theory which focuses on predicting individual players' actions and payoffs and analyzing Nash equilibriums.

Detailed description is beyond the scope of this report.

1.15 Dynamic Game Theory

In game theory, a dynamic game is a game where one player chooses their action before the others choose theirs. Importantly, the later players must have some information of the first's choice, otherwise the difference in time would have no strategic effect. Dynamic games hence are governed by the time axis and represented in the form of decision trees. Detailed description is beyond the scope of this report

1.16 Nash equilibria

In game theory, the Nash equilibria, named after the late mathematician John Forbes Nash, is a proposed solution of a non-cooperative game involving two or more players in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only their own strategy. Detailed description is beyond the scope of this report

1.17 Ito Calculus

Ito calculus, named after Kiyoshi Ito, extends the methods of calculus to stochastic processes such as Brownian motion. It has important applications in stochastic differential equations. The description of Ito Calculus is beyond the scope of this report. However, some results are used to obtain HJB equations.

1.18 Finite Horizon vs Infinite Horizon

When we talk about control of a system, the term “optimal control” makes sense only if we specify the time span of the system operation during which we are concerned about the performance measures.
If we want to control the system, meeting the performance measures for a finite time say T, then the problem is **finite horizon** –

The problem of deriving control $u(t), t = [0, T]$ for the system

$$\dot{x} = Ax(t) + Bu(t)$$

such that the performance index is minimized –

$$PM = \int_0^T x(t)'Qx(t) + u'(t)Ru(t)dt$$

is called a Finite Horizon problem

If we are concerned about the optimality during the whole time span i.e. till $t = \infty$, then it is an **infinite horizon** problem.

The problem of deriving control $u(t), t = [0, \infty]$ for the system

$$\dot{x} = Ax(t) + Bu(t)$$

Such that the performance index is minimized –

$$PM = \int_0^T x(t)'Qx(t) + u'(t)Ru(t)dt$$

is called an Infinite Horizon problem

1.19 Multivariate Gaussian Distribution

Also known as Multivariate Normal Distribution or Joint Normal Distribution is a generalization of the normal distribution to higher dimensions. i.e. a random vector is said to be k-variate normally distributed if every linear combination of its k components has a univariate normal distribution. The multivariate normal distribution of a k-dimensional random vector $X = [X_1, X_2, \dots, X_N]^T$ can be written in the following notation:

$$X \sim \mathcal{N}_k(\mu, \Sigma)$$

where μ is the mean and Σ is the variance and k is the dimension.

1.20 Realization

In systems theory, a realization of a state space model is an implementation of a given input-output behaviour, i.e. given an input-output relationship, a realization of the time-varying matrices $[A(t), B(t), C(t), D(t)]$ is

$$\dot{x(t)} = A(t)x(t) + B(t)u(t)$$

$$y(t) = C(t)x(t) + D(t)u(t)$$

Where $u(t)$ is the input of the system and $y(t)$ is the output

CHAPTER 2
MODEL & PROBLEM FORMULATION

2.1 State, observation and control processes

We consider a dynamical system of *n* controllers, with

State Process: $\{X_t\}_{t=0}^{\infty}, X_t \in \mathcal{X}$

Controller: $i, i \in \{1,2, \ldots n\}$

Observable Process: $\{Y_t^i\}_{t=0}^{\infty}, Y_t^i \in \mathcal{Y}^i$

Control Process: $\{U_t^i\}_{t=0}^{\infty}, U_t^i \in \mathcal{U}^i$

Reward: $\{R_t\}_{t=0}^{\infty}$

The process listed above are related as follows :

1) Let $U_t = \{U_t^1, \ldots, U_t^n\}$ be the control action of all controllers at time *t.* Then, the reward at time *t* depends only of current state X_t, the future state X_{t+1} and the current control actions U_t. Also from the previous result, $\{X_t\}_{t=0}^{\infty}$ is a controlled Markov process **(Proof 1)**, given $\{U_t\}_{t=0}^{\infty}$ and for any $\mathcal{A} \subseteq \mathcal{X}\ and\ \mathcal{B} \subseteq \mathbb{R}$ and any realization $x_{1:t}$ of $X_{1:t}$ and $u_{1:t}$ of $U_{1:t}$, we have

$$\begin{aligned} &P(X_{t+1} \in \mathcal{A}, R_t \in \mathcal{B} |\ X_{1:t} = x_{1:t}, U_{1:t} = u_{1:t}) \\ &\qquad = P(X_{t+1} \in \mathcal{A}, R_t \in \mathcal{B} |\ X_t = x_t, U_t = u_t) \qquad (11) \end{aligned}$$

2) The observations $Y_t = \{Y_t^1, \ldots, Y_t^n\}$ depends only on the current state X_t and the previous control actions U_{t-1} i.e. for any $\mathcal{A}^i \subseteq \mathcal{Y}^i$ any realization $x_{1:t}$ of $X_{1:t}$ and $u_{1:t-1}$ of $U_{1:t-1}$, we have

$$\begin{aligned} &P\left(Y_t \in \prod_{i=1}^{n} \mathcal{A}^i \mid X_{1:t} = x_{1:t}, U_{1:t-1} = u_{1:t-1}\right) \\ &\qquad = P\left(Y_t \in \prod_{i=1}^{n} \mathcal{A}^i \mid X_{1:t} = x_{1:t}, U_{1:t-1} = u_{1:t-1}\right) \end{aligned} \qquad (12)$$

2.2 Information Structure

At time *t,* controller $(i \epsilon \{1, \ldots, n\})$, has access to information I_t^i which is a superset of the history of the history of the observations and control actions at controller i and a subset of the history of the observations and control actions at all controllers, i.e.

$$\{\, Y_{1:t}^i\ , U_{1:t-1}^i\} \subseteq I_t^i \subseteq \{Y_{1:t}, U_{1:t-1}\}$$

The collection $(I_t^i, i \in \{1, \dots, n\}, t = 0,1 \dots)$ is called the information structure of the system and it captures which controller knows what aspect of the system and at what time. A decentralized system is thus characterized by its information structure.

2.3 Control Strategies and Problem Formulation

Based on the information I_t^i available to the controller, controller i chooses action U_t^i using a control law $g_t^i | \, I_t^i \mapsto U_t^i$. The collection of control laws $g^i = \left(g_o^i, g_1^i \dots\right)$ is called the *control strategy of the controller* i. The collection $g = (g^1, \dots, g^n)$ is called the *control strategy of the system*.

The optimization objective is to pick a control strategy g to maximize the expected discount reward.

$$\Lambda(g) = E^g\left[\sum_{i=0}^{\infty} \beta^t R_t\right]$$

for a given discount factor $\beta \in (0,1)$.

2.4 Relationship to Other Models

The decentralized control problem formulated is very similar to dynamic cooperative games. The only key difference is that in the decentralized control all the controllers have a common objective to achieve while in game theory each controller or player has an individual objective. Thus, decentralized control problems are also called *dynamic teams*.

However, decentralized control is conceptually simpler than the corresponding game theory setup –

1) In cooperative game theory, the concepts of bargaining and contracts are used to study when coalitions are formed and how members of the coalition split the value. However, in decentralized control, splitting of the values between the controllers isn't modeled.

2) In dynamic game theory, the concepts of sequential rationality and consistency of beliefs are used to refine the Nash Equilibria. In decentralized stochastic control, all controllers have the same objective hence the conceptual difficulties do not arise.

However, although it offers various advantages over game theory, the optimization problem formulated is non-trivial and the corresponding setup of dynamic cooperative games with incomplete information hasn't yet been discovered.

2.5 Conceptual Difficulties in Finding an Optimal Solution

We notice 2 conceptual difficulties that arise in the optimal design of decentralized stochastic control –

1) It is a functional optimization problem where we have to choose an infinite sequence of control laws *g* to maximize the expected total reward.
2) The domain I_t^i of control laws g_t^i increases with time. Thus, we cannot be certain whether every problem of this nature can be solved or whether the optimal solution can even be implemented.

Although the same difficulties arise in centralized stochastic control, they can be resolved by identifying an appropriate information state process and solving the corresponding dynamic program.

This approach cannot be directly applied to decentralized stochastic control. We need to consider other methods to achieve this same result.

CHAPTER 3
OVERVIEW OF CENTRALIZED STOCHASTIC CONTROL

A centralized stochastic control system is a special case of decentralized stochastic control. Here there is only one controller $(n = 1)$ and the controller has perfect recall $(I_t^1 \subseteq I_{t+1}^1)$ i.e. the controller remembers everything it has seen and done in the past.

The observation, information, control action and control law are given by $Y_t, I_t, U_t\ and\ g_t$ respectively.
Therefore, the information available to the controller at time *t* is given by $I_t^1 = \{\ Y_{1:t}, U_{t:t-1})$.
The action law is given by $g_t = I_t \mapsto\ U_t$to choose a control strategy U_t. The collection of control laws is called a control strategy.

The optimization objective here is to pick a control strategy to maximize the expected discount reward

$$\Lambda(g) = E^g\left[\sum_{i=0}^{\infty} \beta^t R_t\right]$$

for a given discount factor $\beta \in (0,1)$

This model is referred to as a POMDP or a Partially Observable Markov Decision Process. The solution to a POMDP is obtained in 2 steps –

1) We consider a simpler model in which the controller perfectly observes the state of the system i.e. $Y_t = X_t$. Such a model will then be a Markov Decision Process (Proof 1). We then have to show there is no loss of optimality in restricting attention to *Markov Strategies* i.e. control laws of the form $g_t = X_t \longmapsto U_t$. The optimal control strategy of this form is found by solving a dynamic program.

2) A belief system of a POMDP as the *posterior distribution* (see Bayesian Probability) of X_t is defined, given the information i.e. $B_t(\cdot) = P(X_t = \cdot \mid I_t)$. The Belief state is then shown to be a MDP and the results for MDP are then used to solve.

Another approach to this is to identify a *information-state* process of the system and present for the solutions in terms of the information state.

The dynamic program is made using **Definition 1**, **Theorem 1** and **Theorem 2.** This DP is then solved by using various methods such as value iteration or policy-iteration

This information-state based solution approach is equivalent to the standard description of centralized stochastic control. The current state X_t and the belief state $P(X_t = \cdot \mid I_t)$ are respectively the information state in MDP and POMDP

An important property of the information state is that the conditional future reward, which is given by Theorem 1, does not depend on the past and current control strategy $(g_0, g_1 \dots, g_t)$. This *strategy independence* of future cost is critical to obtain a recurrence relation for the conditional future cost as obtained in **Theorem 2**, that does not depend on the current control law g_t.

Based on this recurrence, we can convert the functional optimization problem of finding the best control law g_t into a set of parametric optimization problem of finding the best control action U_t for each realization of the information state Z_t. This resolves the first conceptual difficulty described in section 2.5

CHAPTER 4
CONCEPTUAL DIFFICULTIES IN DYNAMIC PROGRAMMING FOR DECENTRALIZED STOCHASTIC CONTROL

As stated previously in section 2.5, we realize that there are 2 conceptual difficulties that arise in decentralized control. For centralized control, we saw that these difficulties are resolved by identifying appropriate information-state process.

We thus have 2 possible approaches to proceed with decentralized stochastic control –

1) We identify an information state Z_t^i, $Z_t^i \in \mathcal{Z}_t^i$ such that there is no loss of optimality in restricting attention to controllers of the form $g_t^i: Z_t^i \to U_t^i$
   So, essentially, we are finding a set of coupled dynamic programs where each DP is associated with a controller that determines the optimal strategy at that controller.

2) If the probability distribution in the right-hand side of (11) and (12) are time homogenous we can identify a time homogenous information state process and a corresponding dynamic program that determines a time invariant optimal control strategy for all controllers
   Here, essentially, we find a dynamic program that simultaneously determines the optimal control strategy to all controllers.

So, if we consider the first approach, suppose we are able to find a set of coupled dynamic programs, where the dynamic program for controller i, which we refer to as $\mathcal{D}^i$, determines the optimal strategy g^i for controller i. Therefore, the dynamic program $\mathcal{D}^i$ determines the best response strategy g^i for a particular choice of control strategies g^{-i} for other controllers.

$$g^i = \mathcal{D}^i(g^{-i})$$

Any fixed-point $g^* = (g^{*,1}, \ldots, g^{*,n})$ of these coupled dynamic programs has the property that every controller $i, i \in \{1, \ldots, n\}$, is playing its best response strategy to the strategies of other controllers. Such a strategy is called a **Person-by-Person** optimal strategy. One key drawback of the person-by-person optimal strategy is that it may not be globally optimal; in fact, a person-by-person strategy may arbitrarily perform bad as compared to the globally optimal strategy. So, unless we impose further restrictions on the model, a set of coupled dynamic programs cannot determine a globally optimal strategy.

Now, we consider the second approach, suppose we find a dynamic program similar to Theorem 2 that determines the optimal control strategies for all controllers. All controllers must be able to use this dynamic program to find their control strategy. Therefore, the information-state process $\{Z_t\}_{t=0}^{\infty}$ of such a dynamic program must have the following property: Z_t is a function of the information I_t^i available to every controller $i, i \in \{1, \ldots, n\}$. Such a strategy is called a **Common-Information** optimal strategy

These two approaches are analyzed in detail in the next two chapters

CHAPTER 5
PERSON BY PERSON APPROACH

The concept of Person-by-Person approach is very similar to the computational approaches for finding Nash Equilibrium in game theory. It is essentially used in static systems with multiple controllers but have also been used in dynamic systems from time to time.

This approach is used to identify structural results as well as identify coupled dynamic programs to find person-by-person optimal (or equilibrium) strategies.

5.1 Structure of Optimal Control Strategies

The entire procedure can be implemented in 4 steps –

Step 1 – Pick a controller that has perfect recall, say i and arbitrarily fix the control strategies g^{-1} of all the controllers except controller i

Step 2 – Consider the sub-program of finding the best response strategy g^{i} at controller i. Since controller i has perfect recall, the sub-problem is centralized.

Step 3 – Identify an information-state process $\{\tilde{I}_t^i\}_{t=0}^{\infty}$ for this sub-problem. Then, there is no loss of optimality in restricting attention to control laws of the form $\tilde{g}_t^i$: $\tilde{I}_t^i \rightarrow U_t^i$ at controller i.

Step 4 – Repeat this procedure at all controllers that have perfect recall.

Since in Step 1, the choice of control strategies g^{-1} was completely arbitrary, we observe that if the structure of $\tilde{g}_t^i$ does not depend on the choice of control strategies g^{-1} of other controllers. Hence, there is no loss of global optimality in restricting attention to control laws of the form $\tilde{g}_t^i$ at controller i.

5.2 Coupled Dynamic Program for Person-by-Person Optimal Solution

The Person-by-Person Optimal Solution can be determined by a coupled dynamic program. Although on further analysis, we realize that this approach may not always be globally optimal.

In general, the information-state process $\{\tilde{I}_t^i\}_{t=0}^{\infty}$ may not be time-homogeneous.

But even if we suppose that it is time homogenous for every controller i, $i \in \{1, \dots, n\}$, when we arbitrarily fix the control strategies $\tilde{g}^i$ of all other controllers, the dynamical model seen by controller i is not time homogeneous.

For the dynamic model, from the point of view of controller i to be time-homogeneous, we must restrict attention to time-invariant strategies at each controller.
But a time-invariant person-by-person optimal strategy obtained by the coupled dynamic programs need not be globally optimal for two reasons –

First, there might be other time-invariant person-by-person strategies that achieve a higher expected discounted reward.

Second, there might be other time-varying strategies that achieve higher expected discounted reward.

CHAPTER 6
COMMON INFORMATION APPROACH

The common information approach provides a dynamic programming decomposition that determines optimal control strategies for all controllers for decentralized control systems.

This approach works to obtain a dynamic program that determines optimal control strategies for all controllers, such that the information-state process is be measurable at all controllers and, at each step of the dynamic program. A functional optimization problem is solved that determines instructions to map local information to control action for each realization of the information state.

The information available is split at each controller into two parts –

The common information

$$C_t = \bigcap_{\tau \geq t} \bigcap_{i=1}^{n} I_\tau^i$$

And the local information

$$L_t^i = I_t^i \setminus C_t \; \forall \; i \in \{1, \dots, n\}$$

By construction, the common and local information determine the total information i.e.

$$I_t^i = C_t \cup L_t^i$$

And the common information is nested i.e. $C_t \subseteq C_{t+1}$

The common information approach applies to decentralized control systems that have a partial history sharing information structure.

To identify a dynamic program that determines optimal control strategies for all controllers, the common information approach exploits the fact that planning is centralized i.e. the control strategies for all controllers are chosen before the system starts running and therefore optimal strategies can be searched in a centralized manner.

6.1 Basic Model

A Basic Model is first created

1) **The Dynamic System** – Consider a dynamic system with n controllers. The system operates in discrete time for a horizon T. Let $X_t \in \mathcal{X}_t$ denote the state of the system at time t, let $U_t^i \in \mathcal{U}_t^i$ denote the control action of controller $i, (i = 1, \dots, n)$ at time t, and U_t denote the vector $(U_t^1, \dots, U_t^n)$

   The initial state X_1 has a probability distribution Q_1 and evolves according to

$$X_{t+1} = f_t(X_t, U_t, W_t^0)$$

where $\{W_t^0\}_{t=1}^T$ is a sequence of independent and identically distributed random variables with probability distribution Q_W^0.

2) **Data Available to the Controller** – At any time t, each controller has access to three types of data: current observation, local memory, and shared memory.

i) **Current Local Observation**: Each controller makes a local observation $Y_t^i \in \mathcal{Y}_t^i$ on the state of the system at time t

$$Y_t^i = h_t^i(X_t, W_t^i)$$

where $\{W_t^i\}_{t=1}^T$ is a sequence of independently and identically distributed random variables with probability distribution Q_W^i. We assume that the random variables in the collection $\{X_1, W_t^j, t = 1, \dots, T, j = 0,1, \dots, n\}$, called primitive random variables are mutually independent.

ii) **Local Memory**: Each controller stores a subset M_t^i of its past local observations and its past actions in a local memory

$$M_t^i \subset \left\{Y_{1:t-1}^i, U_{1:t-1}^i\right\}$$

at $t = 1$, the local memory is empty, $M_1^i = \phi$

iii) **Shared Memory**: In addition to its local memory, each controller has access to a shared memory. The contents C_t of the shared memory at time t are a subset of the past local observations and control actions of all controllers

$$C_t \subset \{Y_{1:t-1}, U_{1:t-1}\}$$

where Y_t and U_t denote the vectors $(Y_t^1, \dots, Y_t^n)$ and $(U_t^1, \dots, U_t^n)$ respectively. At $t = 1$, the shared memory is empty $C_1 = \phi$

Controller i chooses action U_t^i as a function of the total data (Y_t^i, M_t^i, C_t) available to it. Specifically, for every controller $i, i = 1, \dots, n$

$$U_t^i = g_t^i(Y_t^i, M_t^i, C_t)$$

where g_t^i is called the *control law* of controller i. The collection $g^i = (g_1^i, \dots, g_T^i)$ is called the *control strategy* of controller i. The collection $g^{1:n} = (g^1, \dots, g^n)$ is called the *control strategy* of the system.

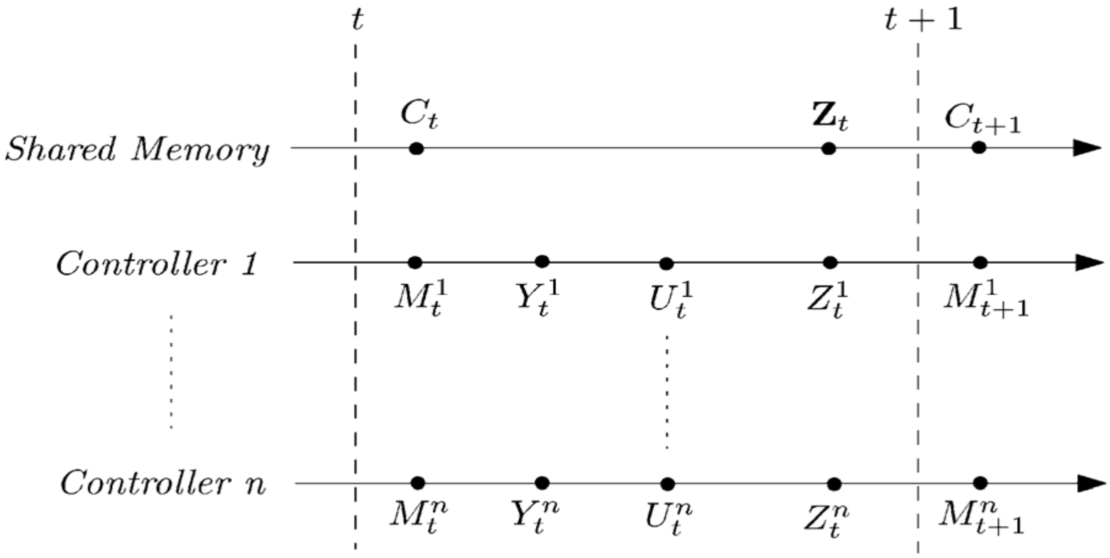
3) **Update of Local and Shared Memories**

i) **Shared Memory Update**: After taking the control action at time t, the local information at controller i consists of the contents M_t^i of its local memory, its local observation Y_t^i and its control action U_t^i. Controller i sends a subset Z_t^i of this local information $\{M_t^i, Y_t^i, U_t^i\}$ to the shared memory. The subset Z_t^i is chosen according to a pre-specified protocol. The contents of shared memory are nested in time, that is, the contents C_{t+1} of the shared memory at time $t+1$ are the contents C_t at time t augmented with the new data $Z_t = (Z_t^1, \dots, Z_t^n)$ sent by all the controllers at time t:

$$C_{t+1} = \{C_t, Z_t\}$$

ii) **Local Memory Update**: After taking the control action and sending data to the shared memory at time t, controller i updates its local memory according to a pre-specified protocol. The content M_{t+1}^i of the local memory can at most equal the total local information $\{M_t^i, Y_t^i, U_t^i\}$ at the controller. However, to ensure that the local and shared memories at time $t+1$ don't overlap, we assume that

$$M_{t+1}^i \subset \left\{M_t^i, Y_t^i, U_t^i\right\} \setminus Z_t^i$$



The figure shows the time order of observations, actions and memory updates. The above model is referred to as the *partial history sharing information structure*

4) **Optimization Problem** – At time t, the system incurs a cost $l(X_t, U_t)$. The performance of the control strategy of the system is measured by the expected total cost

$$J(g^{1:n}) = E^{g^{1:n}}\left[\sum_{t=1}^{T} l(X_t, U_t)\right]$$

where the expectation is with respect to the joint probability measure on $(X_{1:T}, U_{1:T})$ induced by a choice of $g^{1:n}$

A structural result and a dynamic programming decomposition for the decentralized stochastic control problem with partial information sharing is formulated above.

6.2 Equivalent Centralized Stochastic Control Model

The main idea of the proof is to formulate an equivalent centralized stochastic control problem; solve the equivalent problem using classical stochastic-control techniques; and translate the results back to the basic model. This entire procedure can be implemented in 5 steps.

Step 1 – Formulate a centralized coordinated system from the point of view of a coordinator that observes only the common information among the controllers in the basic model

Step 2 – Show that the coordinated system is a POMDP

Step 3 – For the coordinated system, determine the structure of an optimal coordination strategy and a dynamic program to find an optimal coordination strategy.

Step 4 – Show that any strategy of the coordinated system is implementable in the basic model with the same value of the total expected cost. Conversely, any strategy of the basic model is implementable in the coordinated system with the same value of the total expected cost. Hence prove that the two systems are equivalent.

Step 5 – Translate the structural results and dynamic programming decomposition of the coordinated system to the basic model

We can pick a control strategy h to minimize the total cost

$$\Lambda(h) = E^g\left[\sum_{t=1}^{T} l(X_{t,}U_t)\right]$$

where l is the cost function.

We now try to solve the basic problem formulated using the common-information approach as per the steps described above.

Step 1 – Coordinated System

Consider a coordinated system that consists of a coordinator and *n* p/assive controllers. The coordinator knows the shared memory C_t at time t but not the local memories ($M_t^i, i = 1, \dots, n$) or local observations ($Y_t^i, i = 1, \dots, n$). At each time t, the coordinator chooses mapping $\Gamma_t^i = \mathcal{Y}_t^i \times \mathcal{M}_t^i \mapsto \mathcal{U}_t^i, i = 1,2, \dots, n$ according to

$$\Gamma_t = d_t(\mathrm{C}_t, \Gamma_{1:t-1})$$

where $\Gamma_t = (\Gamma_t^1, \Gamma_t^2 \dots \Gamma_t^n)$. The function d_t is called the *coordination rule* at time t and the collection of functions $d = (\mathrm{d}_1, \dots, \mathrm{d}_T)$ is the *coordination strategy*. The selected Γ_t^i is communicated to controller i at time t.

The function Γ_t^i tells controller i how to process its current local observation and its local memory at time t; for that reason, we call Γ_t^i the *coordinator's prescription* to controller i. Controller i generates an action using its prescription as follows:

$$\mathrm{U}_t^i = \Gamma_t^i\,(Y_t^i, M_t^i)$$

For this coordinated system, the system dynamics, the observation model and the cost are the same as the basic model developed earlier.

As before, the performance of a coordination strategy is measured by the expected total cost

$$\Lambda(h) = E^h\left[\sum_{t=1}^{T} l\big(X_{t,}U_t\big)\right]$$

where the expectation is with respect to a joint measure on $(X_{1:T}, U_{1:T})$ induced by the choice of d.

Step 2 – POMDP Model

A partially observable Markov decision process consists of a state process $S_t \in \mathcal{S}$, an observation process $O_t \in \mathcal{O}$, a control process $A_t \in \mathcal{A}$, $(t = 1,2, \dots, T)$, and a single decision-maker where:

1) The action at time *t* is chosen by the decision-maker as a function of observation and action history i.e.

$$A_t = d_t(O_{1:t}, A_{1:t-1})$$

   where d_t is the decision time rule at t

2) After the action at time *t* is taken, the new state and new observation are generated according to the transition probability rule

$$P(S_{t+1}, O_{t+1} \mid S_{1:t}, O_{1:t}, A_{1:t}) = P(S_{t+1}, O_{t+1} \mid S_t, A_t)$$

3) At each time, an instantaneous cost $\bar{l}(S_t, A_t)$ is incurred

4) The optimization problem for the decision-maker is to choose a decision strategy $d = (\mathrm{d}_1, \dots, \mathrm{d}_T)$ to minimize a total cost given as

$$E\left[\sum_{t=1}^{T} \bar{l}\big(S_{t,}A_t\big)\right]$$

We now use **Theorem 3** to show that the coordinated system can be viewed as an instance of the POMDP model by defining the state process as $S_t = \{X_t, Y_t, M_t\}$, the observation process as $O_t = Z_{t-1}$, and the action process $A_t = \Gamma_t$.

Lemma 1 implies that the coordinated system is an instance of the POMDP model described in **Theorem 3**

We thus obtain the structure of optimal strategies and a dynamic program decomposition for POMDPs.

Step 3 – Structural Result and Dynamic Program for the coordinated system:

Since the coordinated system is a POMDP,

Theorem 3 gives the structure of the optimal coordination strategies. So, we define coordinator's information state as –

$$\pi_t = P(S_t|Z_{1:t-1}, \Gamma_{1:t-1}) = P(S_t|C_t, \Gamma_{1:t-1})$$

Then by **Proposition 1**, there is no loss of optimality in restricting attention to coordinate rules of the form

$$\Gamma_t = \hat{d}_t(\pi_t)$$

Furthermore, an optimal coordination strategy of the above form can be found using a dynamic program

$$\pi_{t+1} = \eta_t(\pi_t, Z_t, \Gamma_t)$$

where η_t is the nonlinear standard filtering update function (**See Appendix A**).

We denote the space of possible realizations of π_t by $\mathcal{B}_t$. Thus,

$$\mathcal{B}_t = \Delta(\mathcal{X}_t \times \mathcal{Y}_t^1 \times \mathcal{M}_t^1 \times \ldots \times \mathcal{Y}_t^n \times \mathcal{M}_t^n)$$

We know that $F(\mathcal{Y}_t^i \times \mathcal{M}_t^i, \mathrm{U}_t^i)$ is the set of all functions from $\mathcal{Y}_t^i \times \mathcal{M}_t^i\ to\ \mathrm{U}_t^i$, then by **Proposition 2**

For $t = T, T-1, \ldots 1$ and for all π_t in $\mathcal{B}_t$ we define

$$V_T(\pi) = inf_{\gamma_t^1,\ldots,\gamma_t^n} E\left[\bar{l}(S_T, \Gamma_T)\middle| \pi_T = \pi, \Gamma_T = (\gamma_t^1, \ldots, \gamma_t^n)\right]$$

where $\gamma_t^i \in F(\mathcal{Y}_t^i \times \mathcal{M}_t^i, \mathrm{U}_t^i), i = 1, \ldots, n$ and

$$V_t(\pi) = inf_{\gamma_t^1,\ldots,\gamma_t^n} E[\bar{l}(S_T, \Gamma_T) + V_{t+1}(\eta_t(\pi_t, Z_t, \Gamma_t))| \pi_T = \pi, \Gamma_T = (\gamma_t^1, \ldots, \gamma_t^n)]$$

where $\gamma_t^i \in F(\mathcal{Y}_t^i \times \mathcal{M}_t^i, \mathrm{U}_t^i), i = 1, \ldots, n$

Then the arg inf at each time step gives the coordinator's optimal prescriptions for the controllers when the coordinator's information state is π.

STEP 4 : Equivalence between the Two Models

We first observe that since $C_s \subset C_t$, for all $s < t$, under any given coordination strategy d, we can use C_t to evaluate the past prescriptions by recursive substitution

For example, the past prescription for t=2,3 can be evaluated as functions of $\mathrm{C}_2, \mathrm{C}_3$ as follows –

$$\Gamma_1 = d_1(C_1) = \tilde{d}_1(\mathrm{C}_2)$$

$$\Gamma_2 = d_2\big(C_{2,}\Gamma_1\big) = d_2\big(C_{2,}\tilde{d}_1(\mathrm{C}_2)\big) = \tilde{d}_2(\mathrm{C}_3)$$

And using **Proposition 3**, we can thus state

$$J(g^{1:n}) = \hat{J}(d)$$

STEP 5 – Structural Result and Dynamic Program for the Basic Model

From the **Proposition 1** and **Proposition 3** we can get the following structural result for the Basic Model.

There exist optimal control strategies of the form

$$U_t^i = \hat{g}_t^i\big(Y_t^i, M_t^i, \Pi_t\big)\,, i = 1,2, \dots, n$$

where Π_t is the conditional distribution on X_t, Y_t, M_t given C_t defined as

$$\Pi_t(x, y, m) = P^{\hat{g}^{1:n}_{1:t-1}}(X_t = x, Y_t = y, M_t = m | C_t)$$

For all possible realizations (x, y, m) of (X_t, Y_t, M_t)

Here Π_t is the **Common Information State.**

Consider a control strategy $\hat{g}^i$ for controller i of the form specified above. The control law $\hat{g}_t^i$ at time t is a function from the space $\mathcal{Y}_t^i \times \mathcal{M}_t^i \times \mathcal{B}_t$ to the space of decisions $\mathcal{U}_t^i$. Equivalently, the control law $\hat{g}_t^i$ can be represented as a collection of functions $\{\hat{g}_t^i(\cdot,\cdot,\pi)\}_{\pi \in \mathcal{B}_t}$, where each element of this collection is a function from $\mathcal{Y}_t^i \times \mathcal{M}_t^i$ to $\mathcal{U}_t^i$. An element $\hat{g}_t^i(\cdot,\cdot,\pi)$ of this collection specifies a control action for each possible realization of $\mathcal{Y}_t^i, \mathcal{M}_t^i$ and a fixed realization π of Π_t. We call $\hat{g}_t^i(\cdot,\cdot,\pi)$ the *partial control law* of controller i at time t for the given realization π of the common information state Π_t.

We now use **Proposition 2** to describe a Dynamic Programming Decomposition of the problem of finding optimal control strategies. This dynamic programming decomposition allows us to evaluate optimal *partial control laws* for each realization π of the common information state in a backward inductive manner.

We now define the functions $V_t \mid \mathcal{B}_t \longmapsto \mathbb{R}$, for $t = T, T-1, \dots, 1$ as –

$$V_T(\pi) = \ inf_{\tilde{\gamma}_T^1, \dots, \tilde{\gamma}_T^n} \, E\left[\bar{l}\left(X_T, \tilde{\gamma}_T^1(\mathcal{Y}_T^1, \mathcal{M}_T^1), \dots, \tilde{\gamma}_T^n(\mathcal{Y}_T^n, \mathcal{M}_T^n)\right) \middle| \, \Pi_T = \pi\right]$$

where $\tilde{\gamma}_T^i \in \ F\left(\mathcal{Y}_T^i \times \mathcal{M}_T^i, \mathrm{U}_T^i\right), 1 \leq i \leq n; \forall \, t < T$ and

$$V_t(\pi) = \ inf_{\tilde{\gamma}_t^1, \dots, \tilde{\gamma}_t^n} \, E\begin{bmatrix} \bar{l}\left(X_t, \tilde{\gamma}_t^1(\mathcal{Y}_t^1, \mathcal{M}_t^1), \dots, \tilde{\gamma}_t^n(\mathcal{Y}_t^n, \mathcal{M}_t^n)\right) \\ + V_{t+1}\left(\eta_t(\pi, \tilde{\gamma}_t^1, \dots, \tilde{\gamma}_t^n, Z_t)\right) \end{bmatrix} \mid \Pi_t = \pi$$

where η_t is defined in Appendix C, Proof 4.

For $t = 1, \dots, T$ and for each $\pi \in \mathcal{B}_t$, an optimal partial control law for controller i is the minimizing choice of $\tilde{\gamma}^i$ in the definition of $V_t(\pi)$. Let $\Psi_t(\pi)$ denote the arg inf of the right-hand side of $V_t(\pi)$ and Ψ_t^i denote its ith component. Then, an optimal strategy is given by

$$\hat{g}_t^i(\cdot, \cdot, \pi) = \Psi_t^i(\pi)$$

CHAPTER 7
COMPARISON BETWEEN COMMON INFORMATION APPROACH AND PERSON-BY-PERSON APPROACH

The common information based approach adopted in Chapter 6 differs from the person-by-person approach in many subtle ways. In particular, the structural result of Theorem 2 cannot be found by the person-by-person approach.

If we fix strategies of all but the ith controller to an arbitrary choice, then it is *not necessarily optimal* for controller j to use a strategy as per Theorem 2. This is because if controller's j's strategy uses the entire common information C_t, then controller i, in general, would need to consider the entire common information to better predict controller j's actions and hence controller i's optimal choice of action may also depend on the entire common information.

The use of common information based approach allowed us to prove that *all controllers can jointly use strategies of the form in Theorem 2 without loss of optimality*.

The coordinated system and coordinator described in the common information approach are only used as a tool to explain the approach. The computations carried out at the coordinator are based on the information known to all controllers. Hence, each controller can carry out the computations attributed to the coordinator. As a consequence, it is possible to describe the above approach without considering a coordinator.

CONCLUSION

The controller’s belief on the current state of the system plays a fundamental role for predicting future costs in centralized stochastic control. Hence, the optimal action at the current time is only a function of current belief on the state.

However, in decentralized problems where different controllers have different information, using a controller’s belief on the state of the system presents two main difficulties:

1) Since the costs depend both on system state as well as other controllers’ actions any prediction of future costs must involve a belief on system state as well as some means of predicting other controllers’ actions.
2) Since different controllers have different information, the beliefs formed by each controller and their predictions of future costs cannot be expected to be consistent.

Decentralized stochastic control gives rise to new conceptual challenges as compared to centralized stochastic control. There are two solution methodologies to overcome these challenges:

(1) The **Person-by-Person** approach which provided the structure of globally optimal control strategies and coupled dynamic programs we can determine person-by-person optimal control strategies.

(2) The **Common-Information** approach which provides the structure of globally optimal control strategies as well as a dynamic program that determines globally optimal control strategies and a functional optimization problem that needs to be solved to solve the dynamic program.

In practice, both the person-by-person approach and the common information approach need to be used in tandem to solve a decentralized stochastic control problem as neither approach gives a complete solution on its own

The Person-by-Person approach is first used to simplify the information structure of the system. Then, the common-information approach is used to find a dynamic programming decomposition.

Therefore, a general solution methodology for decentralized stochastic control is as follows.

1. Use the Person-by-Person approach to simplify the information structure of the system.

2. Use the Common-Information approach on the simplified information structure to identify an information-state process for the system.

3. Obtain a dynamic program corresponding to the information-state process.

4. Now we can either obtain an exact analytic solution of the dynamic program (as in the centralized case) or obtain an approximate numerical solution of the dynamic program

The above methodology applies only to systems with partial-history sharing and to systems that reduce to partial-history sharing by a person-by-person approach.

This report hence summarizes the work done by Aditya Mahajan and Mehnaz Menon in the paper “Decentralized Stochastic Control”. It analyzes all the results presented in this paper and provides an in-depth analysis of the proofs of all the models and concepts presented.

The report also looks at the basic concepts mentioned in the paper that are part of optimal stochastic control though not specifically explained in the paper and provides definitions and proofs as and when required.

APPENDIX A (Definitions)

DEFINITION 1 – Information State Process

A process $\{Z_t\}_{t=0}^{\infty}$, is called an information state process If it satisfies the following properties –

1) Z_t is a function of the information I_t available at time *t* i.e. there exists a series of functions $\{f_t\}_{t=0}^{\infty}$ such that

$$Z_t = f_t(I_t)$$

2) The process Z_t is a controlled Markov process controlled by $\{U_t\}_{t=0}^{\infty}$ i.e. for any $\mathcal{A} \subseteq \mathcal{Z}_{t+1}$ and any realization i_i of I_t and any choice u_t of U_t, we have

$$P(Z_{t+1} \in \mathcal{A} \mid I_t = i_t, U_t = u_t) = P(Z_{t+1} \in \mathcal{A} \mid Z_t = f_t(i_t), U_t = u_t) \text{ (7)}$$

3) Z_t absorbs all the available information on the current reward i.e. for any $\mathcal{B} \subseteq \mathbb{R}$, and any realization i_t of I_t and any choice of u_t of U_t, we have

$$P(R_t \in \mathcal{B} \mid I_t = i_t, U_t = u_t) = P(R_t \in \mathcal{B} \mid Z_t = f_t(i_t), U_t = u_t) \qquad (8)$$

DEFINITION 2 – Partial History Sharing

An information structure is called partial history sharing when the following conditions are satisfied –

1) For any set of realizations $\mathcal{A}$ of L_{t+1}^i and any realizations c_t of C_t, l_t^i of L_t^i, u_t^i of U_t^i and y_{t+1}^i of Y_{t+1}^i, we have

$$\begin{gathered} P\left(L_{t+1}^i \in \mathcal{A} \middle| C_t = c_t, L_t^i = l_t^i, U_t^i = u_t^i, Y_{t+1}^i = y_{t+1}^i\right) \\ = P\left(L_{t+1}^i \in \mathcal{A} \middle|, L_t^i = l_t^i, U_t^i = u_t^i, Y_{t+1}^i = y_{t+1}^i\right) \end{gathered}$$

2) The size of the local information is uniformly bounded i.e. there exists a k such that for all t and all $i \in \{1, \dots, n\}$, $\left|\mathcal{L}_t^i\right| \leq k$, where $\mathcal{L}_t^i$ denotes the space of realizations of L_t^i.

   Here the information is split into two parts

   Common Information → $C_t = \bigcap_{\tau \geq t} \bigcap_{i=1}^{n} I_\tau^i$
   Local Information → $L_t^i = I_t^i \setminus C_t \; \forall \, i \in \{1, \dots, n\}$

   By construction we observe, $I_t^i = C_t \cup L_t^i$ and the common information is nested i.e. $C_t \subseteq C_{t+1}$

DEFINITION 3 – Partial Evaluation

For any function $f: (x, y) \longmapsto z$ and a value x_0 of x, the *partial evaluation* of f and $x = x_0$ is a function $g: y \longmapsto z$ such that for all values of y,

$$g(y) = f(x_0, y)$$

APPENDIX B (Theorems)

THEOREM 1 (Structure of Optimal Control Laws)

Let $\{Z_t\}_{t=0}^{\infty}$, $Z_t \in \mathcal{Z}_t$ be an information state process. Then,

1. The information state absorbs the effect of available information on the expected future rewards i.e. for any realizations i_t of the information state I_t, any choice u_t of U_t and any choice of future strategy $g_{(t)} = (g_{t+1}, g_{t+2} \dots)$, we have

$$E^{g_{(t)}}\left[\sum_{\tau=t}^{\infty} \beta^{\tau} R_{\tau} \mid I_t = i_t, U_t = u_t\right] = E^{g_{(t)}}\left[\sum_{\tau=i}^{\infty} \beta^{\tau} R_{\tau} \mid Z_t = f_t(i_t), U_t = u_t\right]$$

2. Therefore, Z_t is a sufficient statistic for performance evaluation and there is no loss of optimality in restricting attention to control laws of the form $g_t: Z_t \longmapsto U_t$

THEOREM 2 (Dynamic Programming Decomposition)

Assume that the probability Distribution in the right hand side of (1),(2),(7) and (8) are time homogenous. Let $\{Z_t\}_{t=0}^{\infty}$ be an information state process such that the space of realization of Z_t is time-invarient i.e. $Z_t \in \mathcal{Z}$

1. For any choice of future strategy $g_{(t)} = (g_{t+1}, g_{t+2} \dots)$, where $g_{\tau}, \tau > t$ is of the form $g_t: Z_t \longmapsto U_t$ and for any realizations z_t of the Z_t and any choice u_t of U_t, we have

$$E^{g_{(t)}}\left[E^{g_{(t+1)}}\left[\sum_{\tau=t+1}^{\infty} \beta^{\tau} R_{\tau} \mid Z_{t+1}, U_{t+1} = g_{t+1}(Z_{t+1})\right] \mid Z_t = z_t, U_t = u_t\right]$$
$$= E^{g_{(t)}}\left[\sum_{\tau=i+1}^{\infty} \beta^{\tau} R_{\tau} \mid Z_t = z_t, U_t = u_t\right]$$

2. There exists a time-invariant optimal strategy $g^* = (g^*, g^*, \dots)$ that is given by

$$g^*(z) = \arg\sup_{u \in \mathcal{U}} Q(z, u) \, , \forall\, z \in \mathcal{Z}$$

where Q is the solution of the following dynamic program

$$Q(z, u) = E[R_t + \beta V(Z_{t+1}) | \, Z_t = z_t, U_t = u_t] \, , \forall\, z \in \mathcal{Z}, u \in \mathcal{U}$$

$$V(z) = sup_{u \in \mathcal{U}} Q(z, u) \; \forall\, z \in \mathcal{Z}$$

THEOREM 3 – POMDP Result

Let θ_T be the conditional probability distribution of the state S_T at time t given the observation $O_{1:t}$ and actions $A_{1:t-1}$

$$\Theta_T(s) = P(S_t = s | O_{1:t}, A_{1:t-1}), s \in \mathcal{S}$$

Then:

1) $\Theta_{t+1} = \eta_t(\Theta_t, A_t, O_{t+1})$ where η_t is the standard non-linear filter: If θ_t, a_t, $o_{t+1}$are the realizations of Θ_t, A_t and O_{t+1}, and $P(s, o|s', a')$ denotes $P(S_{t+1} = s, O_{t+1} = o | S_t = s',\ A_t = a')$, then the realization of the s^{th} element of the vector Θ_{t+1} is

$$\theta_{t+1}(s) = \frac{\sum_{s'} \theta_t(s')\, P(s, o_{t+1}|s', a_t)}{\sum_{\hat{s},\tilde{s}} \theta_t(\hat{s})\, P(\tilde{s}, o_{t+1}|\hat{s}, a_t)} = \eta_t^s(\theta_t, a_t, o_{t+1})$$

   and $\eta_t(\theta_t, a_t, o_{t+1})$ is the vector $(\eta_t^s(\theta_t, a_t, o_{t+1}))_{s \in \mathcal{S}}$

2) There exists an optimal decision strategy of the form

$$A_t = \hat{d}_t(\Theta_t)$$

   Further such a strategy can be found by the following dynamic program:

$$V_T(\theta) = \inf_a E\ [\tilde{l}(\ S_T, a) | \Theta_T = \theta]$$

   and for $1 \leq t \leq T - 1$

$$V_t(\theta) = inf_a\ E\big[\tilde{l}(S_t, a) + V_{t+1}\big(\eta_t(\theta_t, a_t, O_{t+1})\big) \mid \Theta_t = \theta, A_t = a\big]$$

LEMMA 1

For the coordinated system –

1) There exist functions $\widetilde{f_t}$ and $\widetilde{h_t}, t = 1, \ldots, T$ such that

$$S_{t+1} = \widetilde{f_t}(S_t, \Gamma_t, W_t^0, W_{t+1})$$

   and

$$Z_t = \widetilde{h_t}(S_t, \Gamma_t)$$

   In particular, we have

$$P(S_{t+1}, Z_t | S_{1:t}, Z_{1:t-1}, \Gamma_t) = P(S_{t+1}, Z_t | S_t, \Gamma_t) \tag{20}$$

2) Furthermore, there exists function $\tilde{l}$ such that

$$l(X_t, U_t) = \tilde{l}(S_t, \Gamma_t)$$

   Thus, the objective of minimizing can be rewritten as

$$\hat{J}(d) = E\left[\sum_{t=1}^{T} \bar{l}\big(S_{t,}\Gamma_t\big)\right]$$

This lemma is proved as **PROOF 2**

APPENDIX C (Proofs)

PROOF 1 – Markov Process

Let $\{\mathcal{F}_t, t \geq 0\}$ be the canonical filtration of $\{X_t, t \geq 0\}$. For independent increments at any t and $h \geq 0$, the random variable $X_{t+h} - X_t$ is independent of $\mathcal{F}_t$. Then to show that $\{X_t\}$ is a Markov process we know to show that

$$E[f(X_{t+h}|\mathcal{F}_t] = E[f(X_{t+h}|X_t]$$

for any bounded measurable function f on $(\mathcal{S}, \mathcal{B})$ (where $\mathcal{S}$ is the state space and $\mathcal{B}$ is its Borel σ-field). Then,

$$E[f(X_{t+h}|\mathcal{F}_t] = E[f(X_{t+h} - X_t + X_t|\mathcal{F}_t] = E[f(X_{t+h} - X_t + X_t|X_t]$$

Where the last equality implies that X_t is $\mathcal{F}_t$-measurable but $X_{t+h} - X_t$ is independent of $\mathcal{F}_t$.

Hence $\{X_t, t \geq 0\}$ is a Markov Process

PROOF 2 – Lemma 1

The existence of $\widetilde{f_t}$ follows from (1), (2), (7), (10) and the definition of S_t.
The existence of $\widetilde{h_t}$ follows from the fact that Z_t^i is a fixed subset of $\{M_t^i, Y_t^i, U_t^i\}$, (10) and the definition of S_t.
Equation (20) follows from (18) and the independence of W_t^0, W_{t+1} from all random variables in the conditioning in the left-hand side of (20).
The existence of $\tilde{l}$ follows from the definition of S_t and (10)

PROOF 3 – Proposition 2

For any given control strategy $g^{1:n}$ in the basic model, we define a coordinated strategy d for the coordinated system as

$$d_t(C_t) = \left(g_t^1(\cdot,\cdot, C_t), \dots, g_t^n(\cdot,\cdot, C_t)\right)$$

We now fix a specific realization of the primitive random variables $\{X_1, W_t^j, t = 1, \dots T, j = 0,1, \dots, n\}$. We now observe that the realizations will be the same for both the basic model and the coordinated model. Then by the choice of d the realization of the control actions can be made the same. This implies that the realization of the next state and the memories will be the same in the two problems. Thus, the total expected cost in the basic model is the same as the total expected cost under the coordinated strategy i.e.

$$J(g^{1:n}) = \hat{J}(d)$$

PROOF 4 – Update Function η_t of the Coordinator’s Information State

Consider a realization c_{t+1} of the shared memory C_{t+1} at time $t+1$. Let $(\gamma_{1:t})$ be the corresponding realizations of the coordinator’s prescription until time t. We assume the realization $(c_{1:t}, \pi_{1:t}, \gamma_{1:t})$ to be of non-zero probability. Then, the realization π_{t+1} of Π_{t+1} is given by

$$\pi_{t+1}(s) = P\{S_{t+1} = s_t |\ c_{t+1}, \gamma_{1:t}\} \qquad (62)$$

Use **Lemma 1** to simplify the above expression as

$$\sum_{s_t, w_t^0, w_{t+1}} \left[l_s \left(\tilde{f}_t(s_t, \gamma_t, w_t^0, w_{t+1}) \right) . P\{W_t^0 = w_t^0\} \, . P\{W_{t+1} = w_{t+1}\} \, . P\{S_t = s_t | c_{t+1}, \gamma_{1:t}\} \right]$$
(63)

Since $c_{t+1} = (c_t, z_t)$, we write the last term of (63) as

$$P\{S_t = s_t | c_{t+1}, \gamma_{1:t}\} = \frac{P\{S_t = s_t, Z_t = z_t | c_t, \gamma_{1:t}\}}{\sum_{s'} P\{S_t = s', Z_t = z_t | c_t, \gamma_{1:t}\}} \qquad (64)$$

Use **Lemma 1** we write the numerator as

$$P\{S_t = s_t, Z_t = z_t | c_t, \gamma_{1:t}\} = \ l_{\tilde{h}_t(s_t, \gamma_t)}(z_t) \, . P\{S_t = s_t | c_t, \gamma_{1:t}\}$$
(65)

We can drop γ_t from the conditioning above because under the coordinator strategy, it is a function of the rest of terms in the conditioning

$$P\{S_t = s_t, Z_t = z_t | c_t, \gamma_{1:t}\} = \ l_{\tilde{h}_t(s_t, \gamma_t)}(z_t) \, . P\{S_t = s_t | c_t, \gamma_{1:t}\}$$
(66)

Finally, we can substitute (63), (64) and (66) back in (62), to get

$$\pi_{t+1}(s) = \ \eta_t^s(\pi_t, \gamma_t, z_t)$$

APPENDIX D (Propositions)

Proposition 1

There is no loss of optimality in restricting attention to coordinate rules of the form

$$\Gamma_t = \hat{d}_t(\pi_t)$$

This is obtained from **Theorem 1**

Proposition 2

This gives a dynamic program for the coordinator's problem. Since the coordinated system is a POMDP, it implies that computational algorithms for POMDPs can be used to solve the dynamic program for the coordinator's problem as well.

Proposition 3

We state that the basic model and the coordinated system are equivalent and get the following results –

a) Given any control strategy $g^{1:n}$ for the basic model, choose a coordination strategy d for the coordinated system of stage 1 as

$$d_t(C_t) = \left(g_t^1(\cdot,\cdot, C_t), \ldots, g_t^n(\cdot,\cdot, C_t)\right)$$

Then $\hat{J}(d) = J(g^{1:n})$

b) Conversely for any coordination strategy for the coordinated system, choose a control strategy for the basic model as

$$g_1^i(\cdot,\cdot, C_t) = d_1^i(C_1)$$

And

$$g_t^i(\cdot,\cdot, C_t) = d_t^i(C_t, \Gamma_{1:t-1})$$

where

$$\Gamma_k = \mathrm{d}_k(\mathrm{C}_k, \Gamma_{1:k-1}), k = 1,2, \ldots t-1$$

And $d_t^i(\cdot)$ gives the coordinator's prescription for the ith controller. Then,

$$J(g^{1:n}) = \hat{J}(d)$$

BIBLIOGRAPHY

Conceptual difficulties in decentralized control

1) Bernstein, et al, *The complexity of decentralized control of Markov decision processes*, MOR 2002
    - All random variables are finite valued
    - Finite horizon setup
    - The problem of finding the best control strategy is in NEXP (NEXP is the set of decision problems that can be solved by a non-deterministic Turing machine using time 2^{n^O})

2) Whittle and Rudge, *The optimal linear solution of a symmetric team control problem* App. Prob. 1974.
    - Infinite horizon dynamical system with two symmetric controllers
    - Linear dynamics, quadratic cost, and Gaussian disturbance
    - A priori restrict attention to linear controllers
    - Best linear controllers not representable by recursions of finite order

3) Witsenhausen, *A counterexample in stochastic optimum control*, SICON 1969.
    - A two step dynamical system with two controllers
    - Linear dynamics, quadratic cost, and Gaussian disturbance
    - Non-linear controllers outperform linear control strategies cannot use Kalman filtering and Riccati equations

Person by person

1) Mahajan, et al, “Static LQG teams with infinite agents”, *Static LQG teams with Countable Infinite Players*, CDC 2013
    - Intermediate step for extending some results in dynamic teams to infinite horizon.
    - Proxy for large scale systems.
    - Under appropriate symmetry and regularity conditions, the optimal strategy for infinite agents is periodic and obtained by solving a finite dimensional system of linear equations.

2) Marschak, “Static Teams with Finite number of agents”, *Elements for a Theory of Teams*, 1955
    - Correlated observations and coupled costs.
    - Static optimization problem.
    - Seeking an optimal off-line design, not an iterated solution with communication between neighbors.

3) Radnar, “Solution to LQG teams”, *Team Decision Problems*, Ann. Math. Statist., 1962
    - Identify sufficient conditions for optimality
        - Sufficient conditions of Global Optimality (GO)

$$\forall(\tilde{g}^1, \dots, \tilde{g}^n): J(g^1, \dots, g^n) \le J(\tilde{g}^1, \dots, \tilde{g}^n)$$

- Sufficient conditions for Person-by-Person optimality (PBPO)

$$\forall(\tilde{g}^1, \dots, \tilde{g}^n): J(g^1, \dots, g^n) \leq J((g^1, \dots, g^n) \leq J\big(\tilde{g}^i, g^{-i}\big)$$

- Show that when Y are jointly Gaussian and the cost is quadratic in (Y, U)

$$(PBPO) \Rightarrow (GO)$$

- Assume all controllers are linear i.e. $U^i = H^i Y^i$
  $(PBPO) =$ set of all n linear equations : $A_{n\times n} h_{n\times 1} = b_{n\times 1}$ where $h_{n\times 1} = vec[H_1, \dots, H_n]$

Therefore, Global Optimality is achieved by solving n linear equations

Common Information Approach

1) Nayyar, Mahajan, Teneketzis, “Common-info approach for delayed sharing”, *Optimal control strategies in delayed sharing information structures*, IEEE TAC 2011
   - Split available information into two parts – Common Information and Local Information
   - Construct an equivalent centralized coordinated system with Observation History, Control Action and Coordination Law
   - Solve the centralized coordinate system for Information State, Structure of Optimal Controller and Appropriate Dynamic Program.

2) Varaiya and Walrand, *On delayed sharing patterns*, IEEE TAC 1978
   - Proved Witsenhausen’s assertion for $k = 1$ (one-step delay).
   - Counterexample to disprove Witsenhausen’s assertion for $k \geq 2$

3) Witsenhausen, *Seperation of Estimation and Control*, IEEE, 1971
   - Proposed as a bridge between centralized and decentralized systems.
   - Asserted structure of optimal control strategies.

Generalizations and Refinement

1) Arabneydi, et al, *Team optimal control of coupled subsystems with mean field sharing*, CDC 2014
   Mean Field Sharing
   - Motivated by smart grids where agents are weakly coupled through the mean-field.
   - Show that under an appropriate symmetry assumption, the solution scales polynomially with n

2) Mahajan, *Optimal decentralized control of coupled subsystems with control sharing*, IEEE TAC 2013.
   Control Sharing
   - Motivated by communication networks where control actions are observed by all agents.
   - Show that under an appropriate conditional independence assumption, the solution scales exponentially with n

3) Nayyar, et al, *Decentralized stochastic control with partial history sharing: A common information approach*, IEEE TAC 2013
Partial history sharing
- Most general system solvable by common-information approach.
- Many existing results in decentralized control are special cases.
- In the worst case, solution scales double exponentially with n

Markov Process

1) Bauerle, et al, “Theory of Finite Horizon Markov Decision Processes”, *Markov Decision Process with Applications to Finance,* Springer, 2011
2) Fragkiadaki, “Markov Decision Processes”*, Deep Reinforcement Learning and Control,* CMU School of Computer Science
3) Givan, Parr, *An Introduction to Markov Decision Processes,* Purdue and Duke University lecture slides
4) Hansen, et al, *An Improved Policy Iteration Algorithm for Partially Observable MDPs*, 1997
   - Represent Policy as Finite-State Controller
   - Determine the value function, based on the current policy
   - Update the value function, based on Bellman’s equation
   - Update the policy
5) Littman, et al, *Witness Algorithm,* Brown University,1994
   - Start with value vectors for known states
   - Define a linear program (based on Bellman’s equation) that finds a point in
   - the belief space where the value of the function is incorrect
   - Add a new vector (a linear combination of the old value function)
   - Iterate
6) Pineau et al, *Point-Based Value Iteration*, CMU Robotics Institute
7) Puterman, *Markov Decision Processes Discrete Stochastic Dynamic Programming,* John Wiley & Sons, Inc *(1994,2005)*
8) Shani, et al, “A survey of point-based POMDP solvers”, *Autonomous Agents and Multiagent systems* (2013)
9) Smith, Simmons, *Heuristic Search Value Iteration*
10) Spicksma, *Markov Decision Processes, 2015*
11) Zhang, *Algorithms for partially observed Markov decision processes*, PhD Thesis, Hong Kong University of Science and Technology (2001)

Miscellaneous

1) Bayer, *Discretization of SDEs: Euler Methods and Beyond*, PRisMa Workshop, 2006
2) Bellman, *Dynamic Programming,* Princeton University, 1957
3) Bertsekas, *Dynamic Programming and Optimal Control,* Athena Scientific, 1995
4) Mahajan, *Information Structures in Optimal Decentralized Control,* IEEE Conference, 2012
5) Olshausen, *Bayesian probability theory,* 2004
6) Russel, Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1995
7) Shimkin, “Dynamic Programming – Finite Horizon”, *Learning in Complex Systems*, 2011

8) Shimkin, “Dynamic Programming – Infinite Horizon”, *Learning in Complex Systems*, 2011
9) Sutton, et al, Reinforcement Learning: An Introduction, 2017
10) Todorov, *Optimal Control Theory,* University of California San Diego, 2006
11) van Hande, *Stochastic Calculus, Filtering, and Stochastic Control,* Caltech 2007
12) Yoshikawa, *Decomposition of Dynamic Team Problems*, IEEE Transactions on Autonomous Control, 1978

ACKNOWLEDGEMENTS

The author of this Technical Report is grateful to Prof Waseem Mallik for support provided in choosing the research paper, without whose encouragement and motivation, this report would not have been complete