Lecture 11

Note Title 2/14/2012

Viterbi

$$\underset{\bar{p}}{\arg\max} \prod_{t=1}^{T} f_t(\overline{p[t]})$$

BCJR

$$\underset{x}{\arg\max} \sum_{\bar{p}|_i = x} \prod_{t=1}^{T} f_t(\overline{p[t]})$$

The main contributions of the VA & BCJR are

① The trellis-based representation

$\Rightarrow$ Map codewords to paths.

② Efficient iterative computation.

Q: Can we apply ②, the efficient iterative computation, to other (trellis-like) models?

The 3rd representation of the convolutional codes:

A State-based representation, which is commonly termed "the factor graph representation"

* A convolutional code can be represented as a chain of states.

S_0 $\quad$ S_1 $\quad$ S_2 $\quad$ S_3 ...

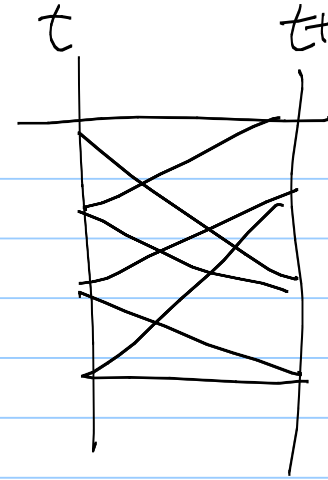
the state of the trellis when $t=0$ $\quad$ when $t=1$ $\quad$ 2 $\quad$ 3

* Each circle reprensents a variable (a state). For comparison, in the trellis representation each circle represents one state value.

* In our running example.

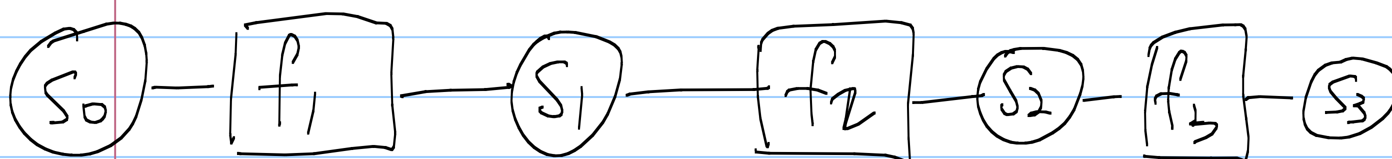
S_0 $\quad$ S_1 $\quad$ S_2 ....

→ It is a variable that has 4 possible values: $\{00, 01, 10, 11\}$.

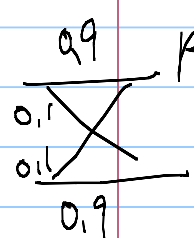


* The trellis structure & the observation $\vec{y}_{obs}$ jointly decides the path metrics $m_{s,s'}$, which is now represented as a function $f_t(s, s')$. And we use a box $\boxed{f_t}$ to represent that function

The new state representation / factor graph becomes:

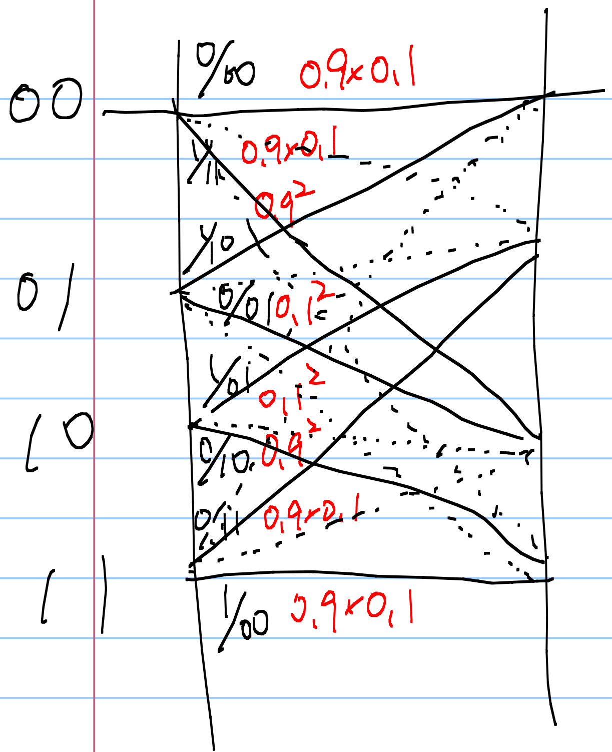


Example: In our running example, consider a BSC channel with cross-over prob $p = 0.1$ & $y_5 y_6 = 10$ then the trellis for the $t=3$ stage is



$t=3$

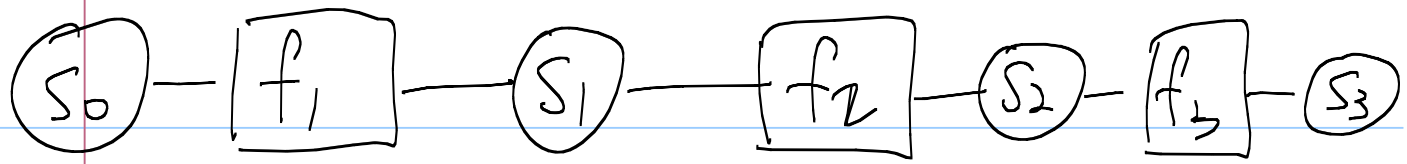
$y_5 y_6 = 10$



$$f_3(S,S') = \begin{cases} 0.9\times 0.1 & \text{if } S=00,\ S'=00 \\ 0.9\times 0.1 & \text{if } S=00,\ S'=10 \\ 0.9^2 & \text{if } S=01,\ S'=00 \\ 0.1^2 & \text{if } S=01,\ S'=10 \\ 0.1^2 & \text{if } S=10,\ S'=01 \\ 0.9^2 & \text{if } S=10,\ S'=11 \\ 0.9\times 0.1 & \text{if } S=11,\ S'=01 \\ 0.9\times 0.1 & \text{if } S=11,\ S'=11 \end{cases}$$

For a simpler representation, we assign $f_3(S,S')=0$ if there is no branch connecting S & S'.

Note again that $f_t(S,S')$ is decided jointly by the observation $\vec{y}$ & the trellis structure



Any set of state values s_1, s_2, s_3 corresponds to a competing candidate of the codeword.

Our goal thus becomes:

$$\max_{\vec{s}} \prod_{t=1}^{\vec{T}} f_t(s_{t-1}, s_t)$$

Viterbi Algorithm

or

$$\operatorname*{argmax}_{x} \sum_{\substack{\vec{s} \text{ with} \\ \text{the } i\text{th} \\ \text{bit being} \\ x}} \prod_{t=1}^{T} f_t(s_{t-1}, s_t)$$

BCJR algorithm

Remark: Since the convolutional code requires zero initial state, we either choose

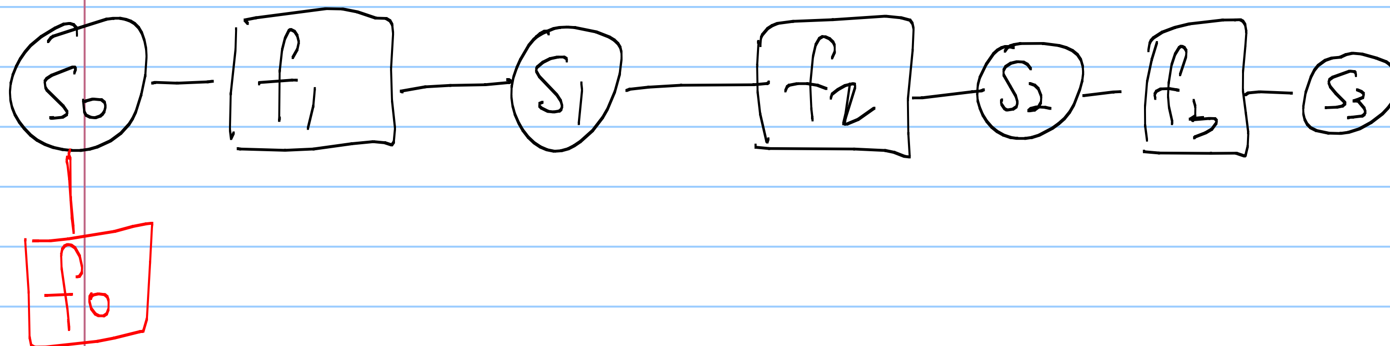
$$\max_{\vec{s}:\, s_0 = 00} \prod_{t=1}^{\vec{T}} f_t(s_{t-1}, s_t)$$

☆ Or we add another function $f_0(S_0)$ such that

$$f_0(S_0) = \begin{cases} 1 & \text{if } S_0 = 00 \\ 0 & \text{otherwise.} \end{cases}$$

This also rules out the unwanted state combinations as those state combinations have 0 objective value.

☆ Then the new representation becomes



& the objective function becomes

$$\max_{\vec{S}} f_0(S_0) \cdot \prod_{t=1}^{T} f_t(S_{t-1}, S_t)$$

or

$$\operatorname*{argmax}_{x} \sum_{\substack{\vec{S} \text{ with} \\ \text{the } i\text{th} \\ \text{bit being} \\ x}} f_0(S_0) \cdot \prod_{t=1}^{T} f_t(S_{t-1}, S_t)$$

Why this state/variable-based representation is called the factor graph representation?

- It "factors" the objective function & presents the relationship between each variable & the factor in a graphical way.

$$f_0(S_0) \cdot \prod_{t=1}^{T} f_t(S_{t-1}, S_t)$$

$\boxed{\text{A detour to factor graphs}}$

* Factor graphs are able to capture a lot of real-world scenarios.

Example: Consider a simple linear code with codeword length 2, and parity-check matrix

$$H = (1, 1)$$

that is $\quad x_1 + x_2 = 0$

ex: $\begin{matrix} 1 & 1 \\ 0 & 0 \end{matrix}$

Both coderwords are equally likely to be chosen.

Suppose we observe $\vec{y}_{obs} = (1, 1)$ through a BSC with cross-over prob p.

Q: Find the factor graph representation of the objective function — the joint prob.

$$P_{\vec{X},\vec{Y}}((\cdot,\cdot),(1,1))$$

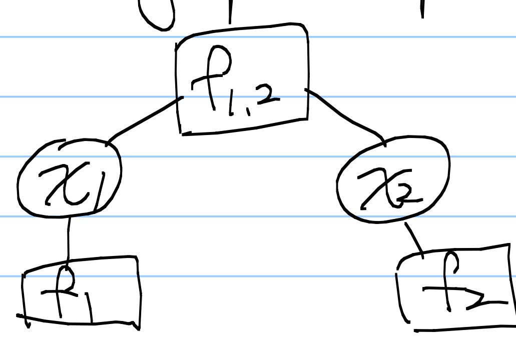
Ans: $P_{\vec{X},\vec{Y}}((\cdot,\cdot),(1,1))$ | Note it depends on the structure & the observation

$$= P_{\vec{X}}(\cdot,\cdot) \cdot P_{\vec{Y}|\vec{X}}((1,1)|(\cdot,\cdot))$$

$$= \underbrace{P_{\vec{X}}(x_1, x_2) \cdot P_{Y_1|X_1}(y_1|x_1) \cdot P_{Y_2|X_2}(y_2|x_2)}_{\text{3 factors.}}$$

2 variables

The factor graph representation becomes



where

$$f_1(\cdot) = P_{Y_1|X_1}(y_1|\cdot)$$

$$f_2(\cdot) = P_{Y_2|X_2}(y_2|\cdot)$$

$$f_{1,2}(x_1, x_2) = \begin{cases} \frac{1}{2} & \text{if } (x_1, x_2) = (0,0),\ (1,1) \iff x_1 + x_2 = 0 \\ 0 & \text{otherwise} \end{cases}$$

---

* Factor graphs are general.

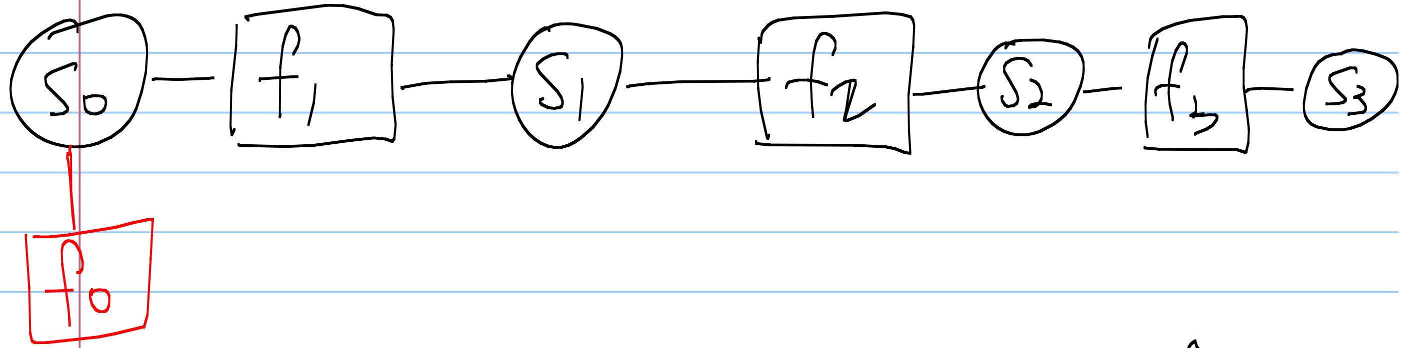
* The Markov inference network is a special case of the factor graph

$$\boxed{P_X} \xrightarrow{X} \boxed{P_{Y|X}} \xrightarrow{Y} \boxed{P_{W|Y}} \xrightarrow{W}$$

$$P_{XYW} = P_X \cdot P_{Y|X} \cdot P_{W|Y}$$

$$\boxed{f_1} - X - \boxed{f_2} - Y - \boxed{f_3} - W$$

Back to our discussion of the convolutional codes



Let us rederive the VA based on the factor graph representation

Obj: $\max_{\vec{S}} f_0(S_0) \cdot \prod_{t=1}^{T} f_t(S_{t-1}, S_t)$

In the VA, the forward metric is $a_s[t]$. Fix $t=t_0$, $a_s[t_0]$ is a function of S. Q: What is the physical meaning of $a_s[t_0]$?

Recall that $a_s[t_0]$ is the value of the optimal partial path from the origin to S at t_0

$$\Rightarrow a_{00}[t_0] = \underset{s_0,\dots,s_{t_0-1}}{\operatorname{argmax}} \; f_0(s_0)\left(\prod_{t=1}^{t_0-1} f_t(s_{t-1},s_t)\right)\cdot f_{t_0}(s_{t_0-1},00)$$

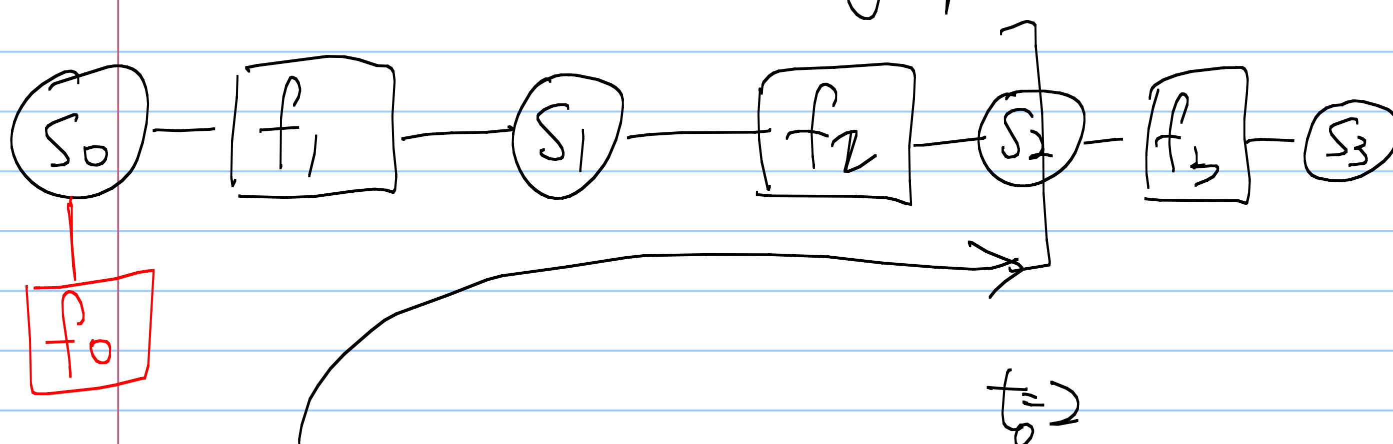
$$a_{01}[t_0] = \underset{s_0,\dots,s_{t_0-1}}{\operatorname{argmax}} \; f_0(s_0)\left(\prod_{t=1}^{t_0-1} f_t(s_{t-1},s_t)\right)\cdot f_{t_0}(s_{t_0-1},01)$$

$$\vdots \qquad\qquad\qquad \vdots$$

$$a_S[t_0] = \underset{s_0,\dots,s_{t_0-1}}{\operatorname{argmax}} \; f_0(s_0)\left(\prod_{t=1}^{t_0-1} f_t(s_{t-1},s_t)\right)\cdot f_{t_0}(s_{t_0-1},S)$$

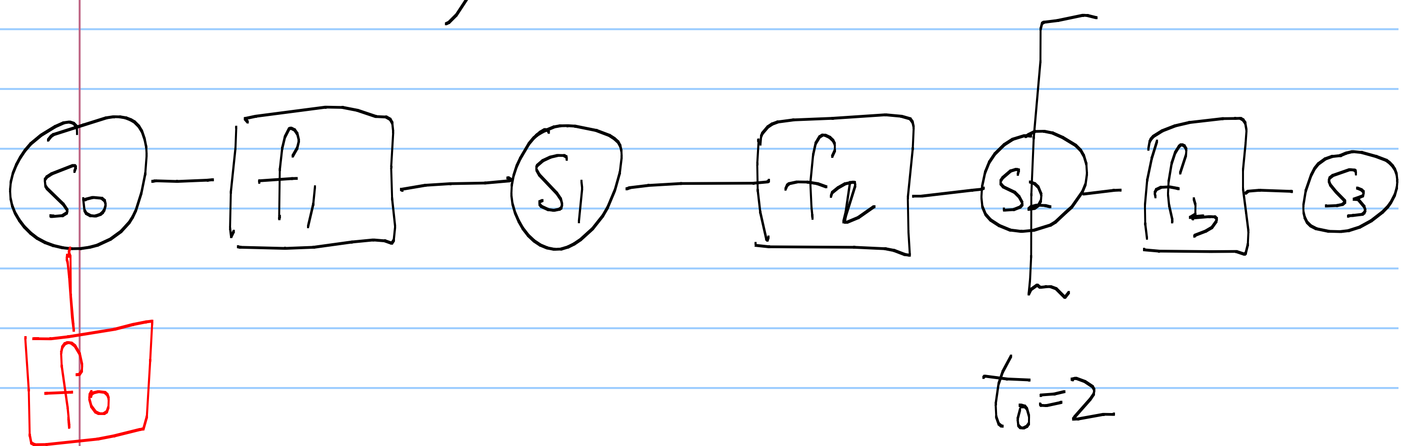
$\triangleq a_{S_{t_0}}$ is a vector of metrics.

$a_S[t_0]$ thus $\boxed{\text{summarizes}}$ the effect of the factor graph on the



left-hand side of S_{t_0} in the factor graph representation.

The 2-way VA is also straightforward



$t_0=2$. $\quad b_s[2] \triangleq b_{s_2} = \max_{s_3} f_3(s_2, s_3) b_{s_3}$

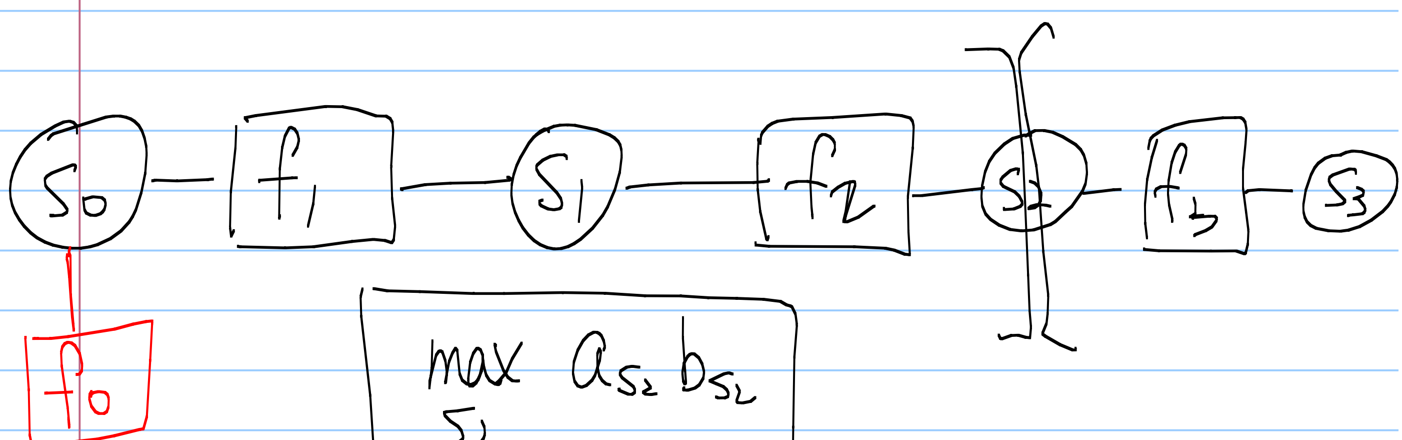
For comparison

$$a_s[2] \triangleq a_{s_2} = \max_{s_1} f_2(s_1, s_2) a_{s_1}$$

$$a_s[0] \triangleq a_{s_0} = \max_{s_0} f_0(s_0)$$

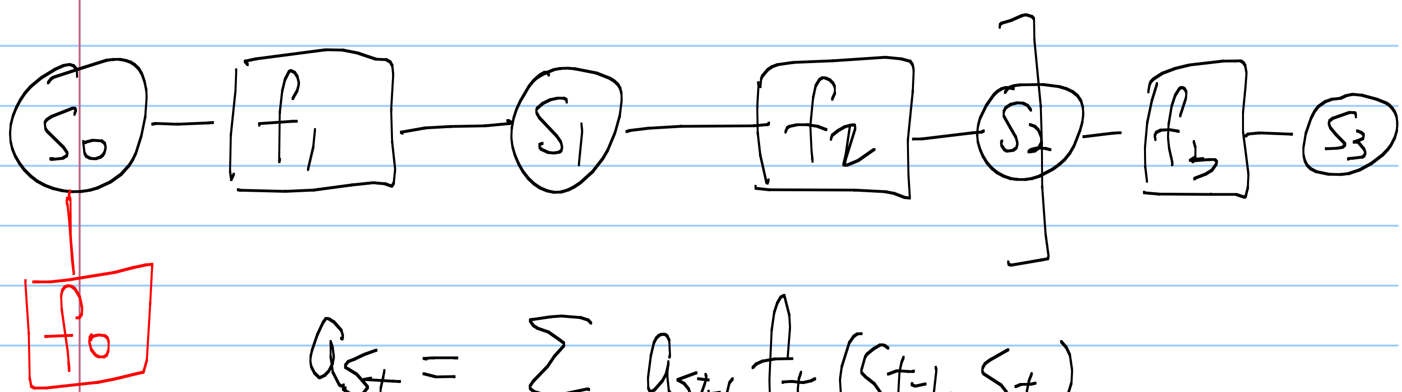
Very straightforward update rules.

To combine the a_{st} & b_{st}.



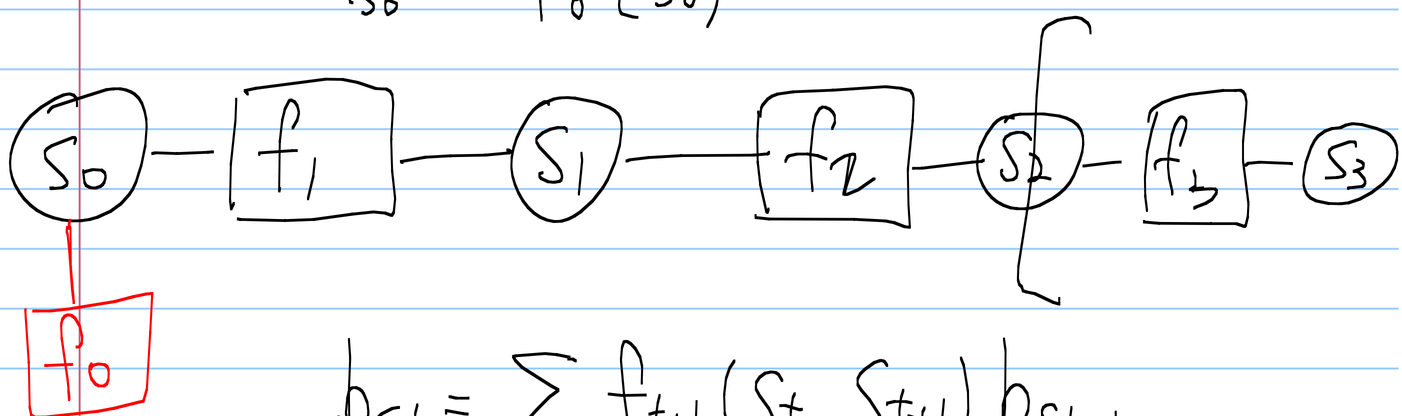
$$\boxed{\max_{s_2} a_{s_2} b_{s_2}}$$

The BCJR algorithm can be derived similarly



$$a_{s_t} = \sum_{s_{t-1}} a_{s_{t-1}} f_t(s_{t-1}, s_t)$$

$$a_{s_0} = f_0(s_0)$$



$$b_{s_t} = \sum_{s_{t+1}} f_{t+1}(s_t, s_{t+1})\, b_{s_{t+1}}$$