Online Supplement for "An Approximation Approach for Response-Adaptive Clinical Trial Design."

Appendix A.1 Table of Notation

Symbol	Definition
n	Total number of patients every period
T	Total number of periods (trial length)
N	Total number of patient observations, $N = n \times T$
J	Set of treatments
O	Set of outcomes
$\mathcal{H}$	Set of health states
$\mathcal{U}$	Set of controls
$\mathcal{K}$	Set of observed outcomes
$\mathbf{h}_t = (h_t^{1,s}, h_t^{1,f}, .., h_t^{\lvert J\rvert,s}, h_t^{\lvert J\rvert,f})$	Health state in period t, $h_t^{j,o} \in [0,1]\ \forall j \in J, o \in \{s,f\}$
$\mathbf{u}_t = (u_t^1, .., u_t^{\lvert J\rvert})$	Controls (actions) in period t
$\boldsymbol{k}_t = (k_t^{1,1}, k_t^{1,f}, .., k_t^{\lvert J\rvert,s}, k_t^{\lvert J\rvert,f})$	Observed outcomes in period t
$(\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t)$	Parameters of the beta distribution
P_t	Transition matrix in period t
p_t^j	Probability of success with treatment $j \in J$
R_t	Rewards in period t
V_t	Value function in period t
Z	Grid resolution
$\tilde{\mathcal{H}}$	Set of approximate health states
$\tilde{\mathcal{X}}$	Set of simplices
$d = \lvert J\rvert \times \lvert O\rvert - 1$	Dimensions of the unit hypercube
$\tilde{\mathbf{h}}_t = (\tilde{h}_t^1, ..., \tilde{h}_t^d, \tilde{h}_t^{d+1})$	Grid state in period t, $\tilde{h}_t^{d+1} = 1 - \sum_{i=1}^{d} \tilde{h}_t^i$
$\bar{\mathbf{h}}_t$	Realized state in period t
λ_t^i	Barycentric coordinates in period t, $\lambda_t^i \in [0,1]$
$\tilde{V}_t$	Approximate value function in period t
$\tilde{Q}_t$	Random variable representing the approximate reward-to-go in period t
$\mu_t(\tilde{Q}_t)$	Mean of the approximate reward-to-go in period t
$\sigma_t^2(\tilde{Q}_t)$	Variance of the approximate reward-to-go in period t
w	Weight on the standard deviation of the approximate reward-to-go
$\xi_t(\tilde{Q}_t)$	Learning-adjusted objective in period t
$\underline{V}_t$	Lower bound on the interpolated value in period t
$\overline{V}_t$	Upper bound on the interpolated value in period t
y_h, y_u	Sampled next state
$\mathcal{H}_h$, $\mathcal{H}_u$	Set of (independently) sampled next states
m_h	Number of outer loop samples
m_u	Number of inner loop samples
$\mathcal{S}^{\pi_i}$	Expected proportion of successes in the trial with design π_i
H	Restricted Horizon
L	Block size for limiting action space
a_m	Preliminary Experiments

Appendix A.2 Proofs

Proof of Proposition 1. We refer the readers to Ahuja and Birge (2016), which proves a similar result for the case of two treatments. □

Proof of Proposition 2. The proof relies on first showing that an optimal decision-maker may expect higher total patient successes in the trial if allowed to observe additional patient outcomes. Formally, we need to show that, for all t, $\boldsymbol{\alpha}_t$, $\boldsymbol{\beta}_t$, and fixed $\mathbf{u} \geq 0$, $\mathbb{E}_{\mathrm{k}}[V_t(\mathbf{h}_t, \boldsymbol{\alpha}_t + \mathbf{k}, \boldsymbol{\beta}_t + \mathbf{u} - \mathbf{k} | \mathbf{u}; \boldsymbol{\alpha}_t; \boldsymbol{\beta}_t)] \geq V_t(\mathbf{h}_t, \boldsymbol{\alpha}_t, \boldsymbol{\beta}_t)$, where the expectation is w.r.to (random outcomes) represented by the vector $\mathbf{k}$. This requires proof by induction and use of Jensen's inequality. The desired result is then obtained by applying this result, use of induction argument and Vandermonde's Convolution. We omit the proof and instead direct the readers to Bertsimas and Mersereau (2007, Section 3.3) for the details, which proves a similar result. □

Proof of Proposition 3. We first note that $\mathcal{M} = \{\mathcal{H}, \mathcal{K}, \mathcal{U}, P_t, R_t\}$ is an MDP; see Bertsekas (1995, Section 5.1) for the proof, which shows that that the addition of the (imperfect) information state preserves the Markovian dynamics. To show that $\tilde{\mathcal{M}}$ is an MDP, we follow similar arguments as in Hauskrecht (2000), which shows that a grid-based approximation of a POMDP preserves the Markovian property, and note that

$$
\begin{aligned}
\tilde{V}_t(\tilde{\mathbf{h}}_t) &= \max_{\mathbf{u}_t} \{\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + \sum_{\mathbf{k}_{t+1} \in \mathcal{K}} P_t(\mathbf{h}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t) \tilde{V}_{t+1}(\mathbf{h}_{t+1})\}. \\
&= \max_{\mathbf{u}_t} \{\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + \sum_{\mathbf{k}_{t+1} \in \mathcal{K}} P_t(\mathbf{h}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t) [\sum_{i=1}^{d+1} \lambda_{t+1}^i \tilde{V}_{t+1}(\tilde{\mathbf{h}}_{t+1}^i)]\} \\
&= \max_{\mathbf{u}_t} \{\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + \sum_{i=1}^{d+1} \tilde{V}_{t+1}(\tilde{\mathbf{h}}_{t+1}^i) [\sum_{\mathbf{k}_{t+1} \in \mathcal{K}} \lambda_{t+1}^i P_t(\mathbf{h}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)]\} \\
&= \max_{\mathbf{u}_t} \{\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + \sum_{i=1}^{d+1} \tilde{V}_{t+1}(\tilde{\mathbf{h}}_{t+1}^i) P_t(\tilde{\mathbf{h}}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)\}
\end{aligned}
$$

where the second line follows from substituting the value of $\tilde{V}_{t+1}(\mathbf{h}_{t+1})$ from (2), and where we denote $\sum_{\mathbf{k}_{t+1} \in \mathcal{K}} \lambda_{t+1}^i P_t(\mathbf{h}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)$ as $P_t(\tilde{\mathbf{h}}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)$ in the last line. Note that $\lambda_t^i \in [0, 1]$, $\sum_{i=1}^{d+1} \lambda_t^i = 1$, and the uniqueness of λ_t^i guarantees that $P_t(\tilde{\mathbf{h}}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)$ can be interpreted as true probabilities. We get the desired result by letting $\mathbb{E}_{\mathbf{k}_{t+1}}[\tilde{V}_{t+1}(\tilde{\mathbf{h}}_{t+1}) = \sum_{i=1}^{d+1} V_{t+1}(\tilde{\mathbf{h}}_{t+1}^i) P_t(\tilde{\mathbf{h}}_{t+1} | \tilde{\mathbf{h}}_t, \mathbf{u}_t)$. □

Proof of Proposition 5. We use the induction argument to prove the first inequality. Consider period $T-1$, the *terminal* decision period, where the optimal strategy is to allocate all patients to the treatment with highest expected success probability, which is the same as lower bound. Thus $V_{T-1} \geq \underline{V}_{T-1}$.

Now assume the inequality holds true for for $t = t+1, ..., T-2$, i.e. $\tilde{V}_{t+1} \geq \underline{V}_{t+1}$. From (1),

$$
\begin{aligned}
V_t &= \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + V_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&\geq \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + \underline{V}_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&\geq \max_{u_t^j=\{0,n\}\forall j\in J} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + \underline{V}_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{u_t^j=\{0,n\}\forall j\in J} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + n(T-(t+1)) \max_{j\in J} [\mathbb{E}p_{t+1}^j|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}] \,|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{u_t^j=\{0,n\}\forall j\in J} \mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t) + n(T-(t+1)) \max_{u_t^j=\{0,n\}\forall j\in J} \mathbb{E}_{\mathbf{k}_{t+1}}[\max_{j\in J} [\mathbb{E}p_{t+1}^j|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}] \,|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= n \max_{j\in J} [\mathbb{E}p_t^j|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] + n(T-(t+1)) \max_{u_t^j=\{0,n\}\forall j\in J} \mathbb{E}_{\mathbf{k}_{t+1}}[\max_{j\in J} \mathbb{E}p_{t+1}^j|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}]|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= n \max_{j\in J} [\mathbb{E}p_t^j|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] + n(T-(t+1)) \max_{j\in J} [\mathbb{E}p_t^j|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= n\,(T-t) \max_{j\in J} [\mathbb{E}p_t^j|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] = \underline{V}_t,
\end{aligned}
$$

where the first inequality follows from the induction argument, the second inequality is due to the fact that optimization over a restricted control space (such that u_t^j is restricted to 0 or n) leads to suboptimal solution, the third equality follows from the fact that $\underline{V}_{t+1}$ is itself a maximum function such that the maximum of sum of two function equals sum of maximums, and the equality in the second-to-last line follows from the fact the expectation in future periods is based on priors in current period.

To prove the second inequality, we again use the induction argument. In period $T-1$, the *terminal* decision period, the optimal strategy is to allocate all patients to the treatment with highest expected success probability. Since $\mathbb{E}p_t^j$ is linear function of α_t^j, Jensen's inequality gives, for a fixed $\sum_{t=0}^{T-1} u_t^j$,

$$
\begin{aligned}
V_{T-1} &= n \max_{j\in J} \mathbb{E}p_{T-1}^j \\
&\leq n\, \mathbb{E} \max_{j\in J} p_{T-1}^j = \overline{V}_{T-1}.
\end{aligned}
$$

Now assume the inequality holds true for $t = t+1, ..., T-2$, i.e. $V_{t+1} \leq \overline{V}_{t+1}$. From (1),

$$
\begin{aligned}
V_t &= \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + V_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&\leq \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + \overline{V}_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{\mathbf{u}_t} [\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + \mathbb{E}_{\mathbf{k}_{t+1}} \overline{V}_{t+1}|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{\mathbf{u}_t} [\mathbb{E}_{\mathbf{k}_{t+1}} R_t(\mathbf{k}_{t+1}) + n(T-(t+1))\, \mathbb{E}_{\mathbf{k}_{t+1}} \mathbb{E} \max_{j \in J} [p^j_{t+1}|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}]|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{\mathbf{u}_t} [\mathbb{E} p^j_t u^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] + n(T-(t+1)) \max_{\mathbf{u}_t} \mathbb{E}\mathbb{E}_{\mathbf{k}_{t+1}} \max_{j \in J} [p^j_{t+1}|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}]|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= \max_{\mathbf{u}_t} [\mathbb{E} p^j_t u^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] + n(T-(t+1))\, \mathbb{E}\max_{j \in J} [p^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&\leq n\mathbb{E} \max_{j \in J} [p^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] + n(T-(t+1))\, \mathbb{E} \max_{j \in J} [p^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] \\
&= n(T-t)\, \mathbb{E}\, [\max_{j \in J} p^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] = \overline{V}_t,
\end{aligned}
$$

where the first inequality follows from the induction argument, the inequality in the second-to-last line follows from Jensen's inequality, and $\mathbb{E}\ \mathbb{E} \max_{j \in J} [p^j_{t+1}|\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1}]|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t] = \mathbb{E}\max_{j \in J} [p^j_t|\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t]$ in the third-to-last line, since the expectation in future periods is based on priors in current period. This concludes the proof. □

Proof of Theorem 1. First note that the computational effort required by π_{Alb} is fundamentally a function of (i) the number of grid points, $|\tilde{\mathcal{H}}|$, which is a function of grid resolution (Z), (ii) the number of actions, $|\mathcal{U}|$, and (iii) the number of outcomes, $|\mathcal{K}|$. This implies that step 1, which requires storing the value function for each grid state and for each period, would require on the order of $(|\tilde{\mathcal{H}}||\mathcal{U}||\mathcal{K}|T)$ operations. Next, we note that each of these operations requires calculating the value of the non-grid points. This requires evaluating each simplex until we find the simplex that contains the non-grid point; this has the computational complexity of $O(|\tilde{\mathcal{X}}|)$. Finally, incorporating the risk measure in the objective requires calculating the variance of the approximate reward-to-go for each grid state, which has the computational complexity of $O(|\mathcal{U}||\mathcal{K}|)$, since the variance must be calculated for each action-outcome pair for each grid state. Combining these we get that the desired result.

We note that calculating the upper bound requires substantial computational effort due to the need to calculate integrals for the beta distribution. However, this can be done offline ahead of time and thus may not add to the computational complexity. □

Appendix A.3 Approximation algorithms

Figure A1: Grid-based approximation algorithm description.

1. **Input**: n, T, J, O, $\boldsymbol{\alpha}_0$, $\boldsymbol{\beta}_0$. **Output**: Value function for each grid state at each period, $\widetilde{V}_t^{lb}(\widetilde{\mathbf{h}}_t, \boldsymbol{\alpha}_t, \boldsymbol{\beta}_t)$.
2. **Determine Grid Resolution**: Choose $Z = 0$. Alternately, determine grid resolution (Z) based on available computational constraints. Establish the grid points using a unit-dimensional hypercube.
3. **Include a learning component in the objective:** Add a weighted measure of the standard deviation to the objective; this learning-adjusted objective is given as follows: $\xi_t(\widetilde{Q}_t) = (1-w)\mu_t(\widetilde{Q}_t) + w\sigma_t(\widetilde{Q}_t)$, and w is the weight on the standard deviation.
4. **Calculate Upper and Lower Bounds**: $\underline{V}_t = n(T-t) \max_{j \in J} \mathbb{E} p_t^j$ and $\overline{V}_t = n(T-t)\, \mathbb{E} \max_{j \in J} p_t^j$, where the expectation is taken with respect to current history (α_t^j, β_t^j).
5. **Solve**: Recursively solve the following equation:
   $\widetilde{V}_t^{lb}(\widetilde{\mathbf{h}}_t) = \max_{\mathbf{u}_t} \xi_t(\widetilde{Q}_t)$, where

   $\widetilde{V}_t^{lb}(\widetilde{\mathbf{h}}_t)$ is the learning-adjusted and bounded value function for the grid-state $\widetilde{\mathbf{h}}_t \in \widetilde{H}$

   $\xi_t(\widetilde{Q}_t) = (1-w)\mu_t(\widetilde{Q}_t) + w\sigma_t(\widetilde{Q}_t)$,

   $\widetilde{Q}_t = R_t(\mathbf{k}_{t+1}) + \widetilde{V}_{t+1}^b(\widetilde{\mathbf{h}}_{t+1})$ is the approximate reward-to-go

   $\mu_t(\widetilde{Q}_t)$ is the mean of this approximate reward-to-go and $\sigma_t(\widetilde{Q}_t)$ is its standard deviation,

   $\widetilde{V}_{t+1}^b = \min\{\overline{V}_{t+1}, \max\{\underline{V}_{t+1}, \widetilde{V}_{t+1}\}\}$ is the bounded value function,

   $\underline{V}_{t+1}$ and $\overline{V}_{t+1}$ are the lower and upper bounds, respectively,

   $\widetilde{V}_{t+1}$ involves calculation of barycentric coordinates, and

   optimal w is determined through cross-validation.

Figure A2: Simulation based approximation algorithm description.

1. **Input**: state ($\widetilde{\mathbf{h}}_t \in \widetilde{\mathcal{H}}$), action space ($\mathcal{U}$), w, n, period $t \in \{0, 1, ..., T-1\}$, $\boldsymbol{\alpha}_0$, $\boldsymbol{\beta}_t$, m_u, and m_h. **Output**: $\widehat{V}_t^{lb}(\widetilde{\mathbf{h}}_t)$

2. **Check**: Can the action-outcome space be enumerated? If yes, find optimal action ($\mathbf{u}_t^*$) by explicitly evaluating all actions and skip to step 4.

3. **Determining $\mathbf{u}_t^*$:**

   Initialize: For each action $\mathbf{u}_t \in \mathcal{U}$, sample outcome sequentially once and set

   $$\widehat{Q}_t(\widetilde{\mathbf{h}}_t, \mathbf{u}_t) = \begin{cases} 0 & \text{if } t = T \text{ and go to step 4,} \\ \widehat{R}_t(\mathbf{r}^T y_u) + \widehat{V}_{t+1}^b(y_u) & if\ t \neq T, \end{cases}$$

   where y_u is the sampled next state (outcome) according to $P_t(\mathbf{h}_{t+1}|\widetilde{\mathbf{h}}_t, \mathbf{u}_t, \boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$, and $\widehat{V}_{t+1}^b = \min\{\overline{V}_{t+1}, \max\{\underline{V}_{t+1}, \widehat{V}_{t+1}\}\}$. Set $\bar{m}_u = 1$.

   Inner loop: Determine optimal action, $\mathbf{u}_t^*$, that achieves

   $\max_{\mathbf{u}_t} \xi_t(\widehat{Q}_t)$, where

   $$\mu_t(\widehat{Q}_t) = \frac{1}{m_u} \sum_{y_u \in \mathcal{H}_u} \widehat{R}_t(\mathbf{r}^T y_u) + \widehat{V}_{t+1}^b(y_u),$$

   $$\sigma_t^2(\widehat{Q}_t) = \mu_t[\widehat{Q}_t^2] - \mu_t^2(\widehat{Q}_t),$$

   $$\xi_t(\widehat{Q}_t) = (1-w)\mu_t(\widehat{Q}_t) + w\sigma_t(\widehat{Q}_t),$$

   where m_u is the number of times the next state has been sampled and optimal w is determined through cross-validation.

   - $\mathcal{H}_u \leftarrow \mathcal{H}_u \cup \{y_u'\}$, where y_u' is the newly sampled next state.
   - Update $\widehat{Q}_t(\widetilde{\mathbf{h}}_t, \mathbf{u}_t)$ with $\widehat{R}_t(\mathbf{r}^T y_u') + \widehat{V}_{t+1}^b(y_u')$ value
   - $\bar{m}_u \leftarrow \bar{m}_u + 1$. If $\bar{m}_u = m_u$ then exit **inner loop**.

4. **Determining optimal value function**: Set $\xi_t(\widehat{Q}_t(\widetilde{\mathbf{h}}_t, \mathbf{u}_t^*), w)$ as

   $$\xi_t(\widehat{Q}_t(\widetilde{\mathbf{h}}_t, \mathbf{u}_t^*), w) = \begin{cases} 0 & \text{if } t = T \text{ and exit,} \\ \xi_t(\widehat{Q}_t(y_h, \mathbf{u}_t^*), w) & if\ t \neq T, \end{cases}$$

   where y_h is the sampled next state (outcome) with respect to $P_t(\mathbf{h}_{t+1}|\widetilde{\mathbf{h}}_t, \mathbf{u}_t^*, \boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$. Set $\bar{m}_h = 1$.

   Outer loop: Set $\widehat{V}_t^{lb}(\widetilde{\mathbf{h}}_t)$ such that

   $$\widehat{V}_t^{lb}(\widetilde{\mathbf{h}}_t) = \frac{1}{m_h} \sum_{y_h \in \mathcal{H}_h} \xi_t(\widehat{Q}_t(y_h, \mathbf{u}_t^*), w)$$

   where m_h is the number of times the next state has been sampled.

   - $\mathcal{H}_h \leftarrow \mathcal{H}_h \cup \{y_h'\}$, where y_h' is the newly sampled next state.
   - Update $\widehat{V}_t^{lb}(\widetilde{\mathbf{h}}_t)$ with $\xi_t(\widehat{Q}_t(y_h', \mathbf{u}_t^*), w)$ value
   - $\bar{m}_h \leftarrow \bar{m}_h + 1$. If $\bar{m}_h = m_h$ then return $\widehat{V}_t^{lb}(\widetilde{\mathbf{h}}_t)$ and exit.

Appendix A.4 Numerical results

Figure A3: Optimality loss (δ_{FAl}) as a function of weight on standard deviation (w).

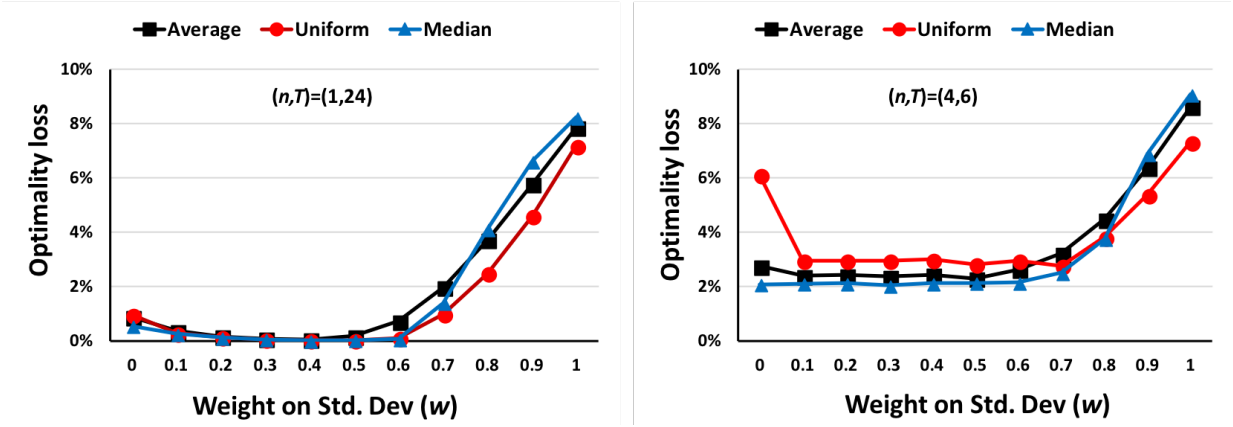

Table A1: Impact of adding bounds on the interpolated value for two small problem instances, Z=0.

			$(n,T)=(1,24)$		$(n,T)=(4,6)$	
w	Bounded	Loss Measure	Average	Uniform	Average	Uniform
0	No	δ_{FA}	1.49%	1.13%	2.66%	6.64%
0	Yes	δ_{FAl}	0.87%	0.96%	2.76%	6.10%
0.4	Yes	δ_{FAlb}	0.05%	0.02%	2.37%	2.96%

Table A2: Optimality loss as a function of grid resolution Z for $(n,T)=(4,6)$.

Z	$\lvert\tilde{\mathcal{H}}\rvert$	$\lvert\tilde{\mathcal{X}}\rvert$	Average	Uniform	CPU-1(sec)	CPU-2(sec)
0	8	5	2.37%	2.96%	0.38	10.07
1	13	20	2.37%	2.96%	2.18	12.66
2	33	80	2.37%	2.96%	13.08	29.90
3	113	320	2.35%	2.92%	79.98	84.98

Notes: $\lvert\tilde{\mathcal{X}}\rvert$ and $\lvert\tilde{\mathcal{H}}\rvert$ represent the number of simplices and grid points, respectively; CPU-1 and CPU-2 represent the computation time for steps 1 and 2, respectively, on 3.5 GHz 6-Core Intel Xeon E5.

Table A3: Performance of π^S_{Alb} with uniform priors for two small problem instances.

		$(n,T)=(1,24)$			$(n,T)=(4,6)$		
m_h	m_u	$\mathcal{S}^{\pi^S_{Alb}}$	Std. Error	δ^S_{FAlb}	$\mathcal{S}^{\pi^S_{Alb}}$	Std. Error	δ^S_{FAlb}
3000	enum	0.6245	0.0006	0.19%	0.5975	0.0006	2.56%
3000	20	0.6181	0.0007	1.22%	0.5797	0.0007	5.47%
3000	50	0.6226	0.0008	0.50%	0.5878	0.0007	4.15%
5000	enum	0.6262	0.0006	-0.08%	0.5958	0.0003	2.84%
5000	20	0.6163	0.0005	1.50%	0.5795	0.0004	5.50%
5000	50	0.6231	0.0005	0.41%	0.5860	0.0005	4.44%

Table A4: Expected proportion of successes ($\mathcal{S}^{Design}$) for π^S_{Alb} and SLAX under various values of H; $(n, T) = (20, 10)$.

$Design \rightarrow$	π^S_{Alb}	SLAX		
		$H = 1$	$H = 2$	$H = 3$
$\mathcal{S}^{Design}$	0.5794*	0.5777	0.5792*	0.5793*
Std. Error	0.00024	0.00026	0.00025	0.00025
CPU	4.534	0.178	0.232	0.514

Notes: CPU represents the computation time (combined for steps 1 and 2) in seconds for a single simulation run.

Table A5: Type I error rates (left) and power values (right) under the null hypothesis of $p^A = p^B = 0.5$ and alternative hypothesis of $p^A = 0.75$ and $p^B = 0.5$ by varying the design parameters Δ and θ_T.

(n,T)=(20,12)

Δ	Null cases - results for the following θ_T				Alternate cases - results for the following θ_T			
	0.75	0.8	0.85	0.9	0.75	0.8	0.85	0.9
0.05	0.1610	0.1290	0.0895	0.0550	0.9930	0.9775	0.9655	0.9435
0.07	0.0965	0.0750	**0.0440**	**0.0280**	0.9795	0.9600	**0.9370**	**0.8895**
0.09	0.0570	**0.0420**	**0.0260**	**0.0110**	0.9445	**0.9290**	**0.8690**	**0.8315**
0.11	**0.0270**	**0.0215**	**0.0110**	0.0070	**0.9075**	**0.8695**	**0.8080**	0.7515
0.13	**0.0110**	**0.0060**	0.0025	0.0020	**0.8395**	**0.8075**	0.7255	0.6490
0.15	0.0045	0.0050	0.0030	0.0010	0.7625	0.6970	0.6270	0.5430

(n,T)=(20,8)

Δ	Null cases - results for the following θ_T				Alternate cases - results for the following θ_T			
	0.75	0.8	0.85	0.9	0.75	0.8	0.85	0.9
0.05	0.2041	0.1485	0.1011	0.0575	0.9812	0.9705	0.9527	0.9174
0.07	0.1099	0.0826	0.0550	**0.0309**	0.9583	0.9373	0.9164	**0.8582**
0.09	0.0663	**0.0431**	**0.0254**	0.0138	0.9282	**0.8936**	**0.8535**	0.7851
0.11	**0.0327**	**0.0240**	0.0132	0.0062	**0.8810**	**0.8441**	0.7882	0.7088
0.13	**0.0168**	0.0128	0.0068	0.0033	**0.8189**	0.7707	0.6981	0.6090
0.15	0.0091	0.0051	0.0026	0.0009	0.7316	0.6852	0.6097	0.4991

Notes: The shaded regions (in grey) represent cells where type I error rate is below 5% (left table) and power of at least 80% (right table). The ones in bold (and shaded striped orange) represent values that meet both type 1 error and power constraints.

Appendix A.4.1 Incorporating learning component in UCB1

We first note that UCB1 index consists of two terms - the current average reward and a second term that represents the confidence bound,[18] ensuring that each action is tried infinitely often while still balancing learning and earning. We tune the UCB1 index by adding a weight to this second term as follows: $UCB1(w)_t^j = (1-w)h_t^j + w\frac{2\log nt}{\sum_{t=0}^{t} u_t^j}$, where UCB1(w) represents the modified index. We calculate the expected proportion of successes with UCB1(w) as a function of the weight on second term (w), where we vary w in increments of 0.1 starting with 0 (see Table A6). We find that there is no observable pattern except for when for $w = 1$, which is significantly worse. Further, the index achieves its highest value at $w = 0.1$ but still remains lower than Greedy and SLAX.

Table A6: UCB1(m): Expected proportion of successes with UCB1(w) as a function of weight on the learning term (w), $Z = 0$.

w	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Mean	0.5719	0.5737	0.5722	0.5730	0.5731	0.5722	0.5722	0.5726	0.5727	0.5728	0.5005
Std. Err	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004

[18] More specifically, the second term is related to the size of the one-sided confidence interval for the average reward within which the true expected reward falls with overwhelming probability (Auer et al., 2002).

Appendix A.4.2 SLAX vs. ADP-BM: regret and computation times

Table A7: Average regret for a variety of simulation problems.

$(\bar{\alpha}^A, \bar{\beta}^A); (\bar{\alpha}^B, \bar{\beta}^B)$	T	n	a_m	Average regret			% Improvement in regret	
							SLAX	SLAX
				Greedy	ADP-BM	SLAX	vs. Greedy	vs. ADP-DM
(1,1); (1,1)	10	20	0	7.68	4.51	1.88	75.5%	58.3%
(1,2); (1,2)	20	20	0	9.10	4.71	2.84	68.7%	39.6%
(1,4); (1,4)	20	20	0	6.81	3.90	2.45	64.0%	37.1%
(2,6); (2,6)	20	50	0	14.41	7.25	4.43	69.2%	38.9%
(10, 6); (6, 6)	10	50	0	8.44	6.25	2.87	66.0%	54.0%
(15,10); (1, 1)	10	100	U[0, 10]	15.91	8.41*	7.48*	53.0%	11.1%
(1,1); (1,1)	20	50	U[0, 20]	7.95*	4.20*	8.00*	-0.7%	-90.4%
(6,2); (6,2)	20	20	U[0, 10]	4.42	3.25	2.62	40.6%	19.4%
(2,6); (2,6)	8	20	U[0, 20]	2.07*	1.91*	1.44	30.3%	24.4%
(2,6); (2,6)	20	50	U[0, 20]	8.47	5.56*	5.38*	36.5%	3.4%
(2,8); (2,8)	6	20	U[0, 20]	1.64*	1.54*	1.01	38.2%	34.1%
(2,50); (2,50)	6	200	U[30, 60]	2.99*	2.85*	2.12	29.0%	25.4%
(2,50); (2,50)	5	1000	U[30, 60]	11.29	9.17	7.42	34.3%	19.1%
(4,100); (4,100)	6	200	U[100, 200]	2.02*	2.07*	1.58	21.6%	23.6%
(4,100); (4,100)	5	1000	U[100, 200]	6.57	6.29	5.27	19.8%	16.2%

Notes: Average regret is calculated as follows: $nT \times (S^{Ideal} - S^{Design})$, *Design*$\in$*{SLAX,Greedy,ADP-BM}*. % improvement in regret by SLAX is defined as follows: $(S^{SLAX} - S^{Design})/(S^{Ideal} - S^{Design})$, *Design*$\in$*{Greedy,ADP-BM}*.

Table A8: Computation times (in seconds) for various methods.

$(\bar{\alpha}, \bar{\beta})$	T	n	a_m	Greedy	ADP-BM	SLAX
2, 6	20	50	0	0.035	2.02	3.673
1, 1	10	20	U[0, 20]	<0.001	0.7484	0.1540
4, 100	5	1000	U[100, 200]	0.001	1.046	2.138

Notes: Numbers represent CPU time (combined for steps 1 and 2) in seconds and are for a single simulation run on a 3.5 GHz 6-Core Intel Xeon E5.

Appendix A.5 Rolapitant trials

Table A9: Details of the rolapitant phase 3 trials.

	MEC	HEC-1	HEC-2	HEC-pooled
Time Frame	Mar 2012-Sep 2013	Feb 2012-Mar 2014		
Total number of patients (N)	1332	544	526	1070
Number of patients per period (n)	36	10	10	20
Trial horizon (T)	37	55	53	54
Allocation to Treatment	666	271	264	535
Allocation to Control	666	273	262	535
Observed Successes on Treatment	475	190	192	382
Observed Successes on Control	410	169	153	322
Treatment Failure rate	28.7%	29.9%	27.3%	28.6%
Control Failure Rate	38.4%	38.1%	41.6%	39.8%

Notes: The terminal period (T) has less than n patients in HEC-1, HEC-2, and HEC-pooled trials.

Table A10: Type I error rates (left) and power values (right) under the null hypothesis of $p^A = p^B = 0.630$ and alternative hypothesis of $p^A = 0.630$ and $p^B = 0.484$ by varying the design parameters Δ and θ_T: MEC (top) and HEC-pooled (Bottom)

MEC

Δ	Null cases - results for the following θ_T				Alternate cases - results for the following θ_T			
	0.75	0.8	0.85	0.9	0.75	0.8	0.85	0.9
0.01	0.9053	0.7781	0.6293	0.4533	1.0000	1.0000	1.0000	1.0000
0.03	0.3361	0.2609	0.1940	0.1217	0.9999	1.0000	0.9993	0.9990
0.05	0.1067	0.0742	**0.0476**	**0.0277**	0.9987	0.9964	**0.9952**	**0.9891**
0.07	**0.0191**	**0.0112**	**0.0069**	**0.0028**	**0.9831**	**0.9780**	**0.9607**	**0.9310**
0.09	**0.0020**	**0.0017**	**0.0006**	0.0003	**0.9186**	**0.8881**	**0.8408**	0.7646
0.11	0.0000	0.0000	0.0001	0.0000	0.7408	0.6724	0.5810	0.4825

HEC-pooled

Δ	Null cases - results for the following θ_T				Alternate cases - results for the following θ_T			
	0.75	0.8	0.85	0.9	0.75	0.8	0.85	0.9
0.01	0.9721	0.8602	0.7138	0.5503	1.0000	1.0000	1.0000	0.9998
0.03	0.4291	0.3421	0.2653	0.1711	0.9996	0.9995	0.9985	0.9963
0.05	0.1604	0.1153	0.0843	0.0519	0.9972	0.9938	0.9884	0.9769
0.07	**0.0433**	**0.0274**	**0.0150**	**0.0078**	**0.9743**	**0.9648**	**0.9445**	**0.8968**
0.09	**0.0077**	**0.0030**	0.0017	0.0009	**0.8971**	**0.8538**	0.7957	0.7017
0.11	0.0006	0.0002	0.0001	0.0000	0.7062	0.6264	0.5484	0.4313

Notes: The shaded regions (in grey) represent cells where type I error rate is below 5% (left table) and power of at least 80% (right table). The ones in bold represent values that meet both type 1 error and power constraints.

Appendix A.6 Objective: learning about treatments

One could argue that the goal of a clinical trial is to identify the most efficacious treatment (instead of maximizing patient successes in the trial), a reason why *fixed* designs are still popular and serve as a gold standard. In other words, the objective should focus on maximizing learning about the treatments. Thus, we implement SLAX under an alternate objective representing the probability of correctly identifying the most efficacious treatment at the end of the trial. Following Ahuja and Birge (2016), we define the reward function under this *learning* objective as follows: $R_T = \max\{Pr(p_T^A > p_T^B), Pr(p_T^B > p_T^A)\}$ and $R_t = 0$ for $t = 0, 1, ..., T-1$. This measure is conceptually similar to predictive probability of success (based on posteriors) that is used to draw statistical inference in Bayesian response-adaptive trials (e.g., Berry et al., 2010).

Letting $\mathcal{P}_t$ denote the value function (V_t) for the learning objective, the dynamic program in (1) can be expressed as follows:

$$\mathcal{P}_T = \max\{Pr(p_T^A > p_T^B), Pr(p_T^B > p_T^A)\}, \text{ and}$$

for $t = 0, 1, ..., T-1$,

$$\mathcal{P}_t(\boldsymbol{\alpha}_t, \boldsymbol{\beta}_t) = \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[\mathcal{P}_{t+1}(\boldsymbol{\alpha}_{t+1}, \boldsymbol{\beta}_{t+1})]. \tag{14}$$

A consequence of the above formulation is that we do not need to store approximate value-to-go functions, as we can directly evaluate the objective based on the observed posteriors. This obviates the need for grid-based approximation and the associated methods to improve approximation accuracy. However, given the problem size, we still require methods for a computationally efficient implementation of SLAX, one that involves adaptive sampling methods, restricted horizon approximation, and action space restriction.

Since a fixed design is primarily focused on learning, any comparison should be benchmarked against it. Consequently, we define a new design called *Equal Allocation* (EA), where patients are allocated to treatments in equal proportion; in our two-armed setting, this implies $n/2$ patients are allocated to each treatment at each period. Let ${\mathcal{P}_t}^{Design}$, $Design \in \{SLAX, EA\}$, represent the expected learning in period t. The following result shows that our proposed adaptive design yields better learning about treatments compared to a fixed design, where patients are allocated equally to each arm.

Proposition 6. ${\mathcal{P}_t}^{SLAX} \geq {\mathcal{P}_t}^{EA}$.

Table A11: Expected probability of identifying the superior treatment (${\mathcal{P}_t}^{Design}$); $(n, T) = (20, 10)$.

	t	1	2	3	4	5	6	7	8	9	10
SLAX	Mean	0.7442	0.8157	0.8505	0.8708	0.8844	0.8946	0.9024	0.9087	0.9139	0.9181
	Std Error	0.0003	0.0004	0.0003	0.0004	0.0004	0.0004	0.0004	0.0004	0.0003	0.0003
EA	Mean	0.7422	0.8111	0.8443	0.8651	0.8796	0.8900	0.8982	0.9049	0.9105	0.9153
	Std Error	0.0003	0.0003	0.0000	0.0002	0.0005	0.0002	0.0002	0.0005	0.0005	0.0005

The proof of the above is intuitive, since equal allocation (EA) is one of the many possible allocations that SLAX allows and is thus subsumed under SLAX. We, thus, omit the proof.

We numerically evaluate the performance of SLAX using the same large problem instance as described in §4.2: $(n, T) = (20, 10)$, uniform starting priors, $m_h = 3000$, and $m_u = 50$, $H = 2$ and $l = 5$ implying $L = 4$. However, instead of the Greedy design, we use EA as the myopic heuristic for the restricted horizon approximation, given that it serves as a gold standard design for learning. We also calculate the expected probability of identifying the superior treatment with EA. Our results, shown in Table A11, are consistent with Proposition 6 . At each period, SLAX is better than EA (all differences are significant at 95% confidence level), where we note that the expecting learning at the start of the trial equals 0.5 for both designs, given uniform starting priors. Further, the gain that SLAX provides over EA increases until period 3 and then decreases, a consequence of the fact that the myopic heuristic is implemented at this period.

While the gains provided by SLAX may appear modest, the realized gains in terms of reduced trial length and/or fewer patients required in the trial could be meaningful in practice, as we demonstrate in Appendix A.6.1. A key additional advantage of SLAX is the opportunity to evaluate multiple objective functions that the current designs do not currently allow, for example, a (weighted) combination of patient successes and learning about treatments.

Appendix A.6.1 Rolapitant Trial: Solution under learning objective

We next implement SLAX under the learning objective, as defined in (14). We calculate the probability of finding the most efficacious treatment at the end ($\mathcal{P}_T$) of each of the four trials (see Table A12 left panel, column: Max), where we assume that starting priors are noninformative (uniform[0,1]), as is the case when using a fixed design.

The leaning objective can be used by trial administrators to conduct interim analysis and stop the trial sooner (or continue it longer), depending on the *stopping criterion* that must be specified in advance (FDA, 2018). The stopping criterion or the *desired probability* of finding the

Table A12: Expected learning and sample size requirements for various rolapitant phase 3 trials (desired $\mathcal{P}_t = 99\%$)

	Uniform starting priors			Informative priors from phase 2		
	Max	Sample size	Saving	Max	Sample size	Saving
MEC	0.9961	1044	21.6%	0.9994	288	78.4%
HEC-1	0.9281	--	--	0.9908	530	3.6%
HEC-2	0.9926	480	9.4%	0.9988	150	71.7%
HEC-pooled	0.9968	780	27.8%	0.9995	240	77.8%

most efficacious treatment is captured in $\mathcal{P}_t$. We also calculate the sample size requirements for each of the four trials when the desired probability is 99%.[19] We notice that in three of the four trials, we achieve the desired probability before the terminal period, resulting in savings that range from 9.4% to 27.8%, depending on the trial and the desired probability. For example, we can conclude rolapitant to be the more efficacious treatment using 21.6% fewer patients than currently used in the MEC trial when the desired probability is 0.99, equivalent to concluding the trial 18 weeks sooner.[20] We also conduct a similar analysis using informative priors (from phase 2). The probability that rolapitant is the more efficacious treatment based on the priors is already very high, at 97.83% and thus, not surprisingly, we see a much larger reduction in sample size requirements (see Table A12, right panel).

Appendix B General model: multiple outcomes

Below, we describe the model for multiple outcomes in detail, where we replace the terms "patients" and "treatments" with unified terms: "subjects" and "interventions," respectively.

Let n and T be the number of subjects per period and the number of periods, respectively, such that $N = nT$ represents the total number of subjects (observations). We note that the first set of randomizations (decisions) take place in period $t = 0$ and the first set of outcomes are observed at time $t = 1$. Unless otherwise specified, decisions are made at the beginning of a period and after observing the results from previous trials. No decisions are made at time $t = T$.

Let J and O be the set of interventions and outcomes, respectively. The number of health conditions is then a Cartesian product of those sets ($J \times O$).

The state of the system is a vector $\mathbf{h}_t \in \mathcal{H} \subseteq \Re^{|J|\times|O|}$, defined as follows:

[19] The *reported* p-values in each of the four trials are 0.0002 (MEC), 0.00006 (HEC-1), 0.0426 (HEC-2) and 0.0001 (HEC-pooled).

[20] The early availability of the drug on the market not only has potential health benefits to patients who need this drug but also potential monetary benefits to the firm, given a longer effective patent life for the drug (number of years remaining in a drug's patent term after the FDA approval).

$$\mathbf{h}_t = (h_t^{1,1}, .., h_t^{|J|,|O|}).$$

Here, $h_t^{j,o} \in \Re_+$ represents the *fraction* of observed subjects to date in health condition (j, o) in period $t \in \{0, 1, .., T\}$, $h_t^{j,o} \in [0, 1]$ for all $j \in J,\ o \in O$, and $|\cdot|$ denotes the cardinality of a set. Further, let $\mathbf{h}_t^{j,\cdot} = (h_t^{j,1}, .., h_t^{j,|O|})$. Since $\sum_{j \in J, o \in O} h_t^{j,o} = 1$, $|J| \times |O| - 1$ components of $\mathbf{h}_t$ are sufficient to fully define the state. The controls (actions), $\mathbf{u}_t \in \mathcal{U} \subseteq \mathbb{Z}_+$, are the number of subjects to allocate to each intervention at each stage, defined as follows:

$$\mathbf{u}_t = (u_t^1, .., u_t^{|J|}).$$

Here, u_t^j is the number of subjects allocated to intervention $j \in J$ in period $t \in \{0, 1, .., T-1\}$, and $\sum_{j \in J} u_t^j = n$. The intervention j is given in period $t-1$, and outcome o is observed in period t. Next, we define the vector of probabilities as follows:

$$\mathbf{p}_t^{j,\cdot} = (p_t^{j,1}, .., p_t^{j,|O|}),$$

where, $p_t^{j,o}$ represents the probability of observing outcome $o \in O$ in period $t+1$ given intervention $j \in J$ in period t. We assume a generalized multinomial likelihood on the transition to state $\mathbf{h}_{t+1}$ from state $\mathbf{h}_t$, given $\mathbf{p}_t$, and then use a Dirichlet conjugate prior on $\mathbf{p}_t$ with hyperparameters $\boldsymbol{\alpha}_t = (\alpha_t^{1,1}, .., \alpha_t^{|J|,|O|})$. Denoting the initial priors by $\boldsymbol{\alpha}_0 = (\alpha_0^{1,1}, ..., \alpha_0^{|J|,|O|})$ and with the assumption that the outcomes of subjects in different health conditions are not informative of each other, each $\alpha_t^{j,o}$ can then be updated independently as follows: $\alpha_t^{j,o} = \alpha_0^{j,o} + nth_t^{j,o}$, where $nth_t^{j,o}$ captures all the (random) realizations from the past for that health condition.

Given $\mathbf{u}_t$, the (random) outcomes are observed in the next period, captured in the vector $\mathbf{k}_{t+1} \in \mathcal{K} \subseteq \mathbb{Z}^{|J| \times |O|}$ such that

$$k_{t+1}^{j,o} = n(t+1)h_{t+1}^{j,o} - nth_t^{j,o}$$

denotes the incremental number of subjects in condition (j, o), where intervention $j \in J$ is given in time t and outcome $o \in O$ is observed in period $t+1$.

Additionally, we define the following terms: $\mathbf{p}_t = (p_t^{1,1}, .., p_t^{|J|,|O|})$, $\boldsymbol{\alpha}_t^{j,\cdot} = (\alpha_t^{j,1}, .., \alpha_t^{j,|O|})$, $\boldsymbol{\alpha}_0^{j,\cdot} = (\alpha_0^{j,1}, .., \alpha_0^{j,|O|})$, and $\boldsymbol{k}_t^{j,\cdot} \equiv (k_t^{j,1}, .., k_t^{j,|O|})$ for all $j \in J$. The entries of the transition matrix in period t representing the probability of transitioning to $\mathbf{h}_{t+1}$ (equivalently realizing $\mathbf{k}_{t+1}$ outcomes) given $\mathbf{h}_t$, $\mathbf{u}_t$, and $\boldsymbol{\alpha}_t$, is then defined as follows:

$$P_t(\mathbf{h}_{t+1}|\mathbf{h}_t, \mathbf{u}_t, \boldsymbol{\alpha}_t) = \prod_{j \in J} Pr(\mathbf{h}_{t+1}^{j,\cdot}|\mathbf{h}_t^{j,\cdot}, u_t^j, \boldsymbol{\alpha}_t^{j,\cdot}) = \prod_{j \in J} \int_0^1 Pr(\mathbf{k}_{t+1}^{j,\cdot}|u_t^j, \mathbf{p}_t^{j,\cdot}) g(\mathbf{p}_t^{j,\cdot}|\mathbf{h}_t^{j,\cdot}, \boldsymbol{\alpha}_t^{j,\cdot}) d\mathbf{p}_t^{j,\cdot},$$

if $u_t^j \in \mathbb{Z}$ and $k_{t+1}^{j,o} \leq u_t^j$ for all $j \in J$, $o \in O$, and zero otherwise. Here, $Pr(\mathbf{k}_{t+1}^{j,\cdot}|\mathbf{h}_t^{j,\cdot}, u_t^j, \boldsymbol{\alpha}_t^{j,\cdot})$ is the multinomial likelihood or the marginal joint distribution and $g(\mathbf{p}_t^{j,\cdot}|\mathbf{h}_t^{j,\cdot}, \boldsymbol{\alpha}_t^{j,\cdot})$ is the probability density function for the Dirichlet distribution.

Finally, the reward, R_t, depends on the objective function. Following existing literature, we initially choose our objective as maximizing outcomes of subjects in the trial. Consequently, R_t is a function of the realized outcomes, such that $R_t(\mathbf{k}_{t+1}|\mathbf{u}_t) = \mathbf{r}^{\mathrm{T}}\mathbf{k}_{t+1}$ for $t \in \{0, 1, ..., T-1\}$ and $R_T = 0$, where $\mathbf{r} \in \Re^{|J|\times|O|}$ represents the vector of rewards for an individual subject in each health condition, independent of the controls applied.

The entire formulation is then a dynamic program in which the objective is to maximize the expected value function (V_t) that captures the expected total reward (i.e.$V_t = \sum_{t=1}^{T} R_t$) and solves the Bellman equation as follows:

$$V_t(\mathbf{h}_t) = \max_{\mathbf{u}_t} \mathbb{E}_{\mathbf{k}_{t+1}}[R_t(\mathbf{k}_{t+1}) + V_{t+1}(\mathbf{h}_{t+1})].$$