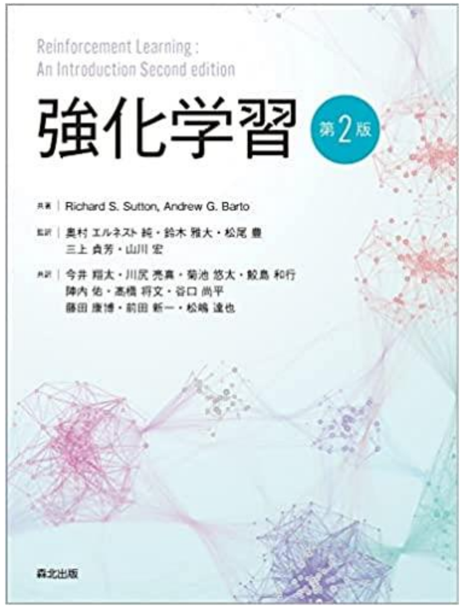
ポケモンで きょうか がくしゅう
1 にゅうもんへん

ポケモンを例題として, ベルマン方程式を理解する

名古屋工業大学 助教 上村知也



はじめまして! きょうか がくしゅうの せかいへ ようこそ!
わたしの なまえは サットン みんなからは きかい がくしゅうの はかせと したわれて おるよ

この せかいでは きょうか がくしゅう と よばれる アルゴリズム たちが
いたるところに つかわれている!

その きょうか がくしゅうを ひとは こうぎょうに つかったり ロボットに つかったり・・・
そして・・・
わたしは この アルゴリズムの けんきゅうを している というわけだ

いよいよ これから
きみの きょうか がくしゅうの がくしゅうの はじまりだ!

ゆめと ぼうけんと!
きょうか がくしゅうの せかいへ!
レッツ ゴー!

もくじ

1. 問題設定

例題：ピカチュウ対コイキング

- ピカチュウとコイキングの戦い
- 問題はゲーム実機よりも簡単にしておく
  - 技は（技に固有の）確率でヒットし, 固定値のダメージを与える
  - 乱数によるダメージの増減は考えない
  - 状態異常や能力値変化は考慮しない, あるいはその影響は無視できる
  - 天候は考慮しない
  - PPは尽きないものとする
  - 道具は使用できない
  - ポケモンは交代できない
  - せいかくや持ち物その他による能力値や状態の変化は発生しない

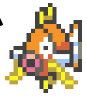
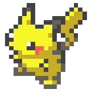
コイキングの設定

- HP：21
- 持ち物なし
- わざ：はねる
  - 攻撃し合うとややこしいので，コイキングは一切反撃できないように技を設定する
  - PP切れにはならないので，わるあがきは出せない

ピカチュウの設定

- HP：（今回関係ない）
- 持ち物なし
- わざ：たいあたり, でんじほう
- たいあたり：命中率100%, ダメージ5
- でんじほう：命中率50%, ダメージ20
  - ダメージ期待値はでんじほうの方が高い

状態と行動

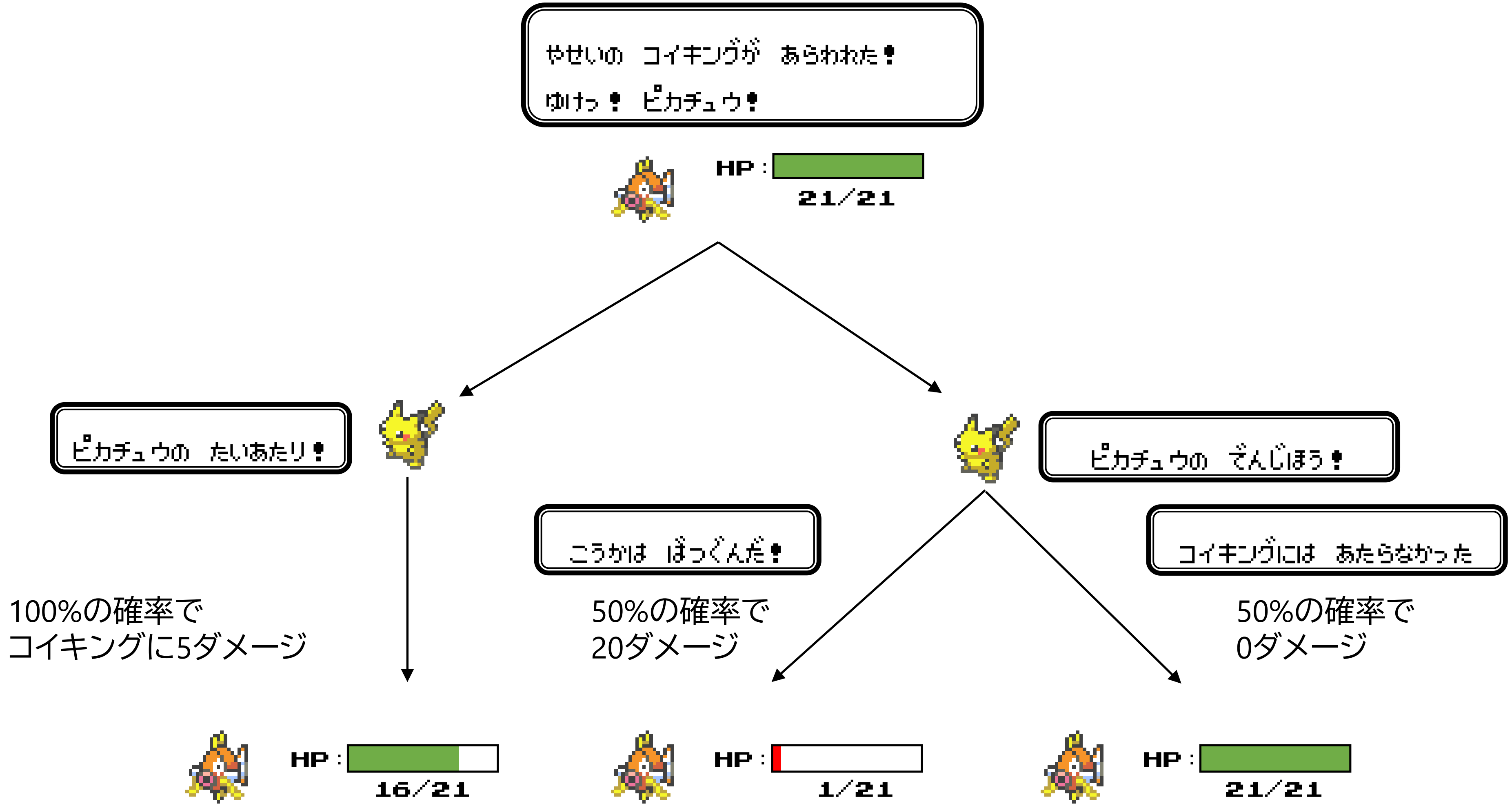
- 状態s_tは, tターン目のコイキングの残りHP
- 行動a_tは, tターン目のピカチュウの技
- ピカチュウは1ターンに1回**行動**を行い(技を使用する), コイキングの**状態**(残りHP)が確率的に変化する
- ピカチュウの攻撃の結果, コイキングの残りHPが0または負の値になったとき, 残りHPをゼロにする
  - このときを**終端状態**として, 試合が終了する

可能な状態の集合 $S = \{0,1,6,11,16,21\}$

可能な行動の集合 $A = \{T, D\}$

T: たいあたり, D:でんじほう

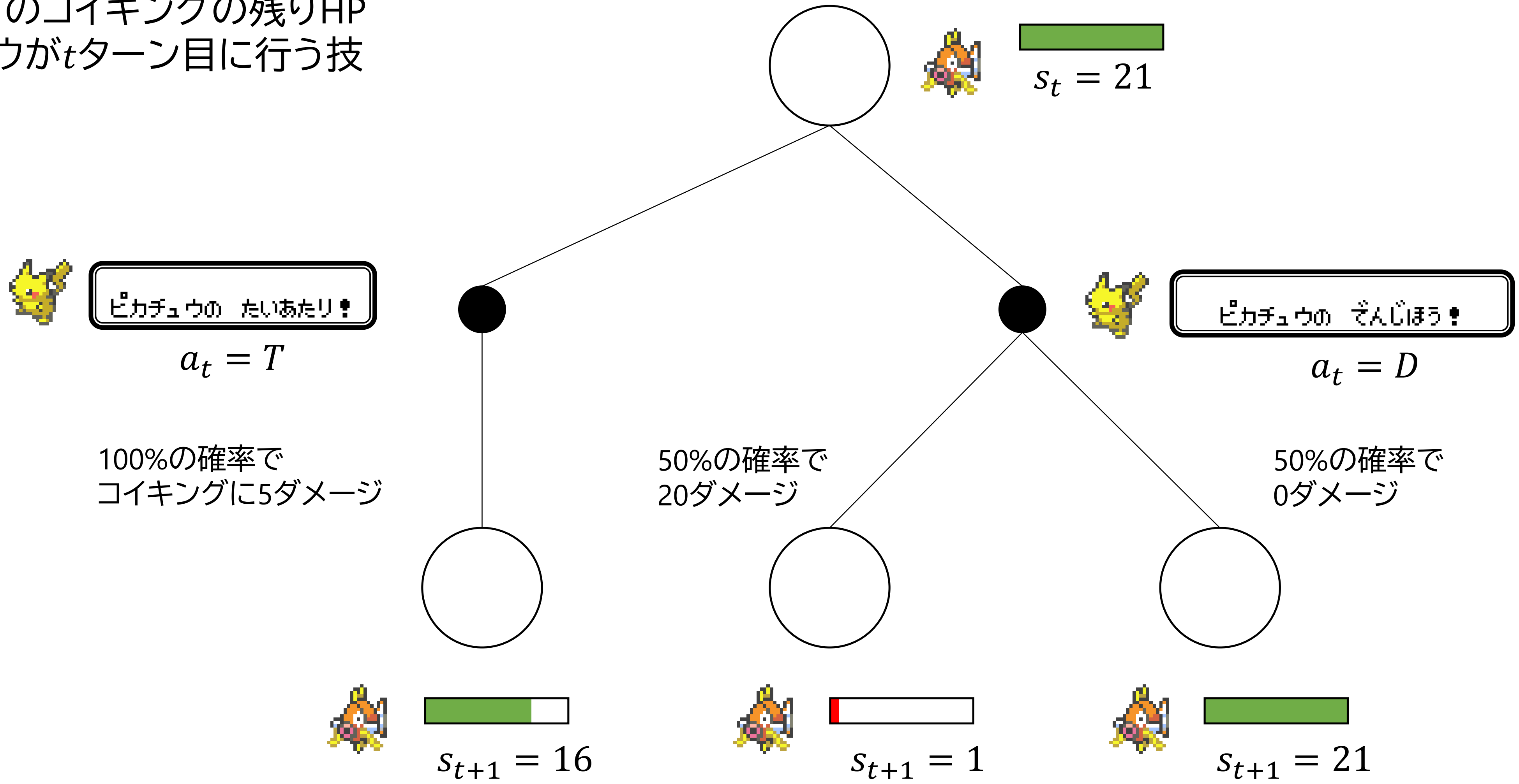
ある状態遷移(1ターン目)



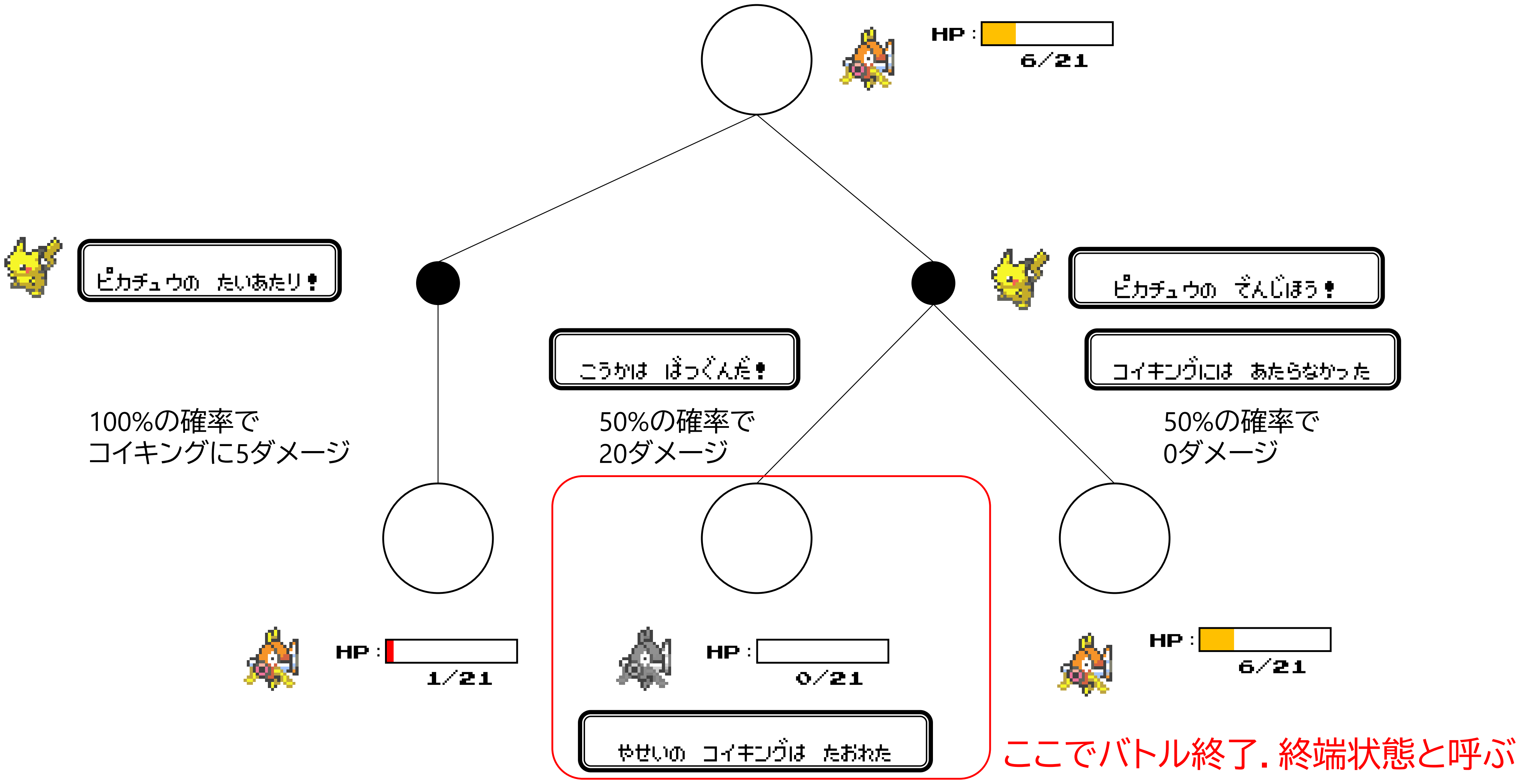
2ターン目のコイキングの残りHPは, ピカチュウの行動に基づいて確率的に決定する

バックアップ線図

- 状態を○, 行動を●で表す
- 状態s_tはtターン目のコイキングの残りHP
- 行動a_tはピカチュウがtターン目に行う技



ある状態遷移(Nターン目)



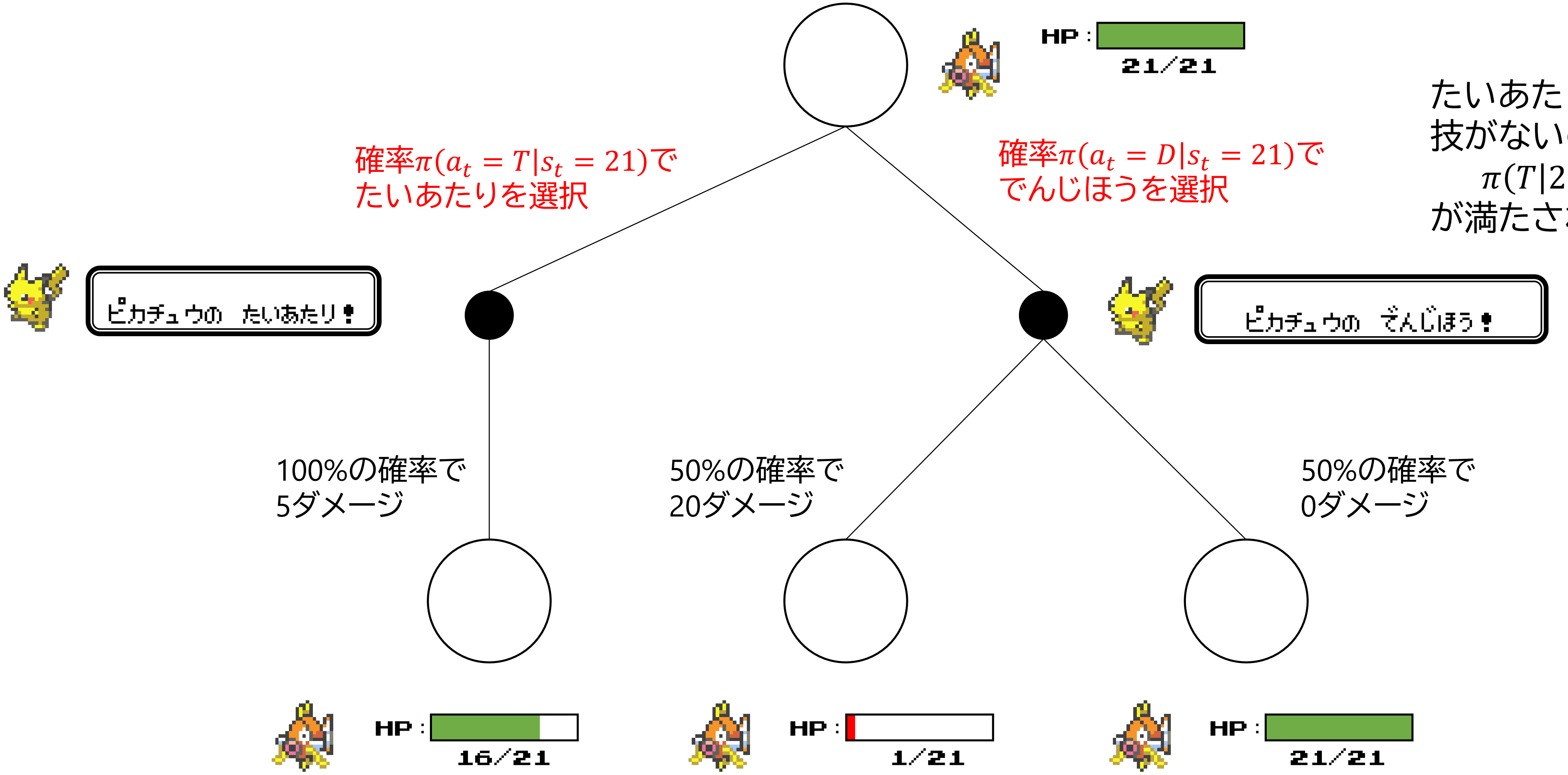
マルコフ決定過程

- 新しい状態s_{t+1}が, その直前の状態s_tと行動a_tのみに依存して（確率的に）決定するとき, そのような遷移を**マルコフ決定過程（Marcov decision process, MDP）**と呼ぶ
- 今回設定した問題において, コイキングの残りHPは, その直前のターンの残りHPとピカチュウの技によって確率的に決定するので, マルコフ決定過程である
  - わざのレパートリーによっては, MDPにならないこともある
  - 例えば, 次のターンの技の威力を2倍にする「じゅうでん」という技を使うと, 状態がその直前のターンだけで決定しなくなってしまう

強化学習問題

- 以上の問題設定で,「ピカチュウが技の選択する指針」を考える
- ここでの「技の選択指針」を, **方策**πと呼ぶ
- $\pi(a_t|s_t)$は現在の状態s_tに対して行動a_tを選択する確率を表す
- 一定のアルゴリズム下で, **最適方策**π^*を導きたい
  - 最適とは, 最小のターン数でコイキングに勝利すること, と定義する

方策πに基づく行動の選択



たいあたりかでんじほうしか技がないので,

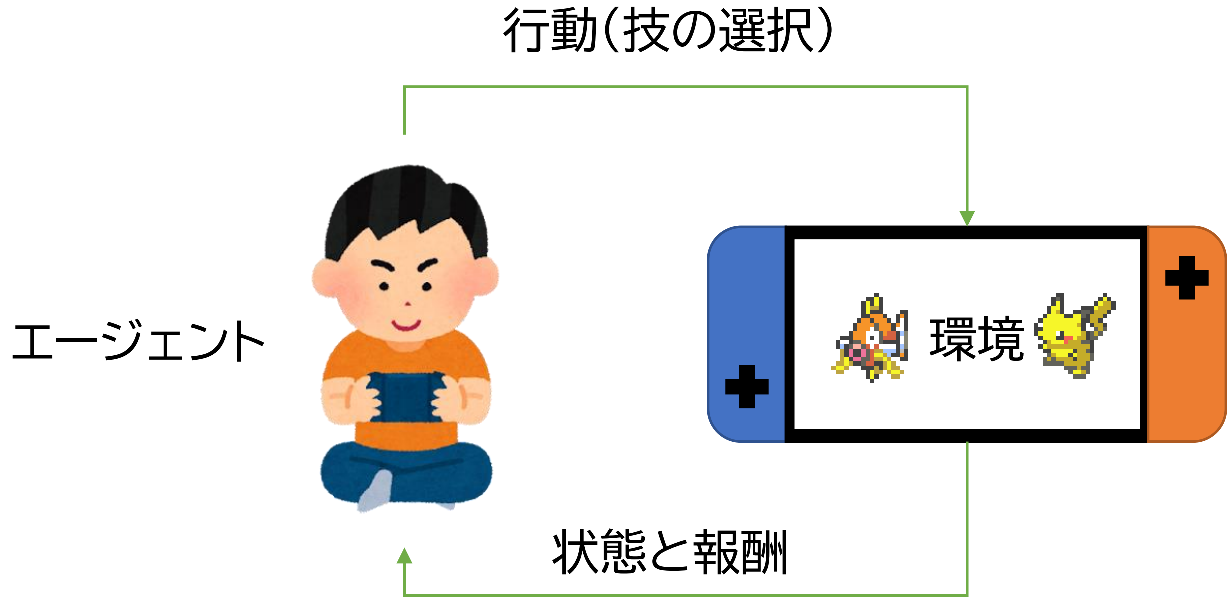
$$\pi(T|21) + \pi(D|21) = 1$$

が満たされる

コラム：エージェントと環境

- 強化学習問題では，エージェントが行動を選択し，環境が変化する
- 環境からは，行動の結果として新しい状態と報酬を得る

問	上記のポケモン問題において，エージェントと環境はそれぞれ何？
誤答	エージェントが「ピカチュウ」で，環境が「コイキングとの戦闘」
正解	エージェントはゲーム「ポケモン」のプレイヤーで，環境はポケモン世界 （ピカチュウもコイキングも技も全て環境に含まれる）



間違いやすいポイント
エージェントは意思決定を行う存在
技を繰り出しているのはピカチュウだが，
その技を選択したのはプレイヤーなので
エージェント＝プレイヤーである

2. 価値関数とベルマン方程式

報酬設計

- 方策の良し悪しを評価するために, **報酬**r_tを設定する
- コイキングに勝利することが目的なので, $s_{t+1}=0$となった瞬間に$r_t=10$を与える
- 長々と戦うことに価値はないので, それ以外の状態では$r_t=-1$を与える
- 戦いが終わったあとは何をしても変わらないので, $r_t=0$を与える
- 累計の報酬

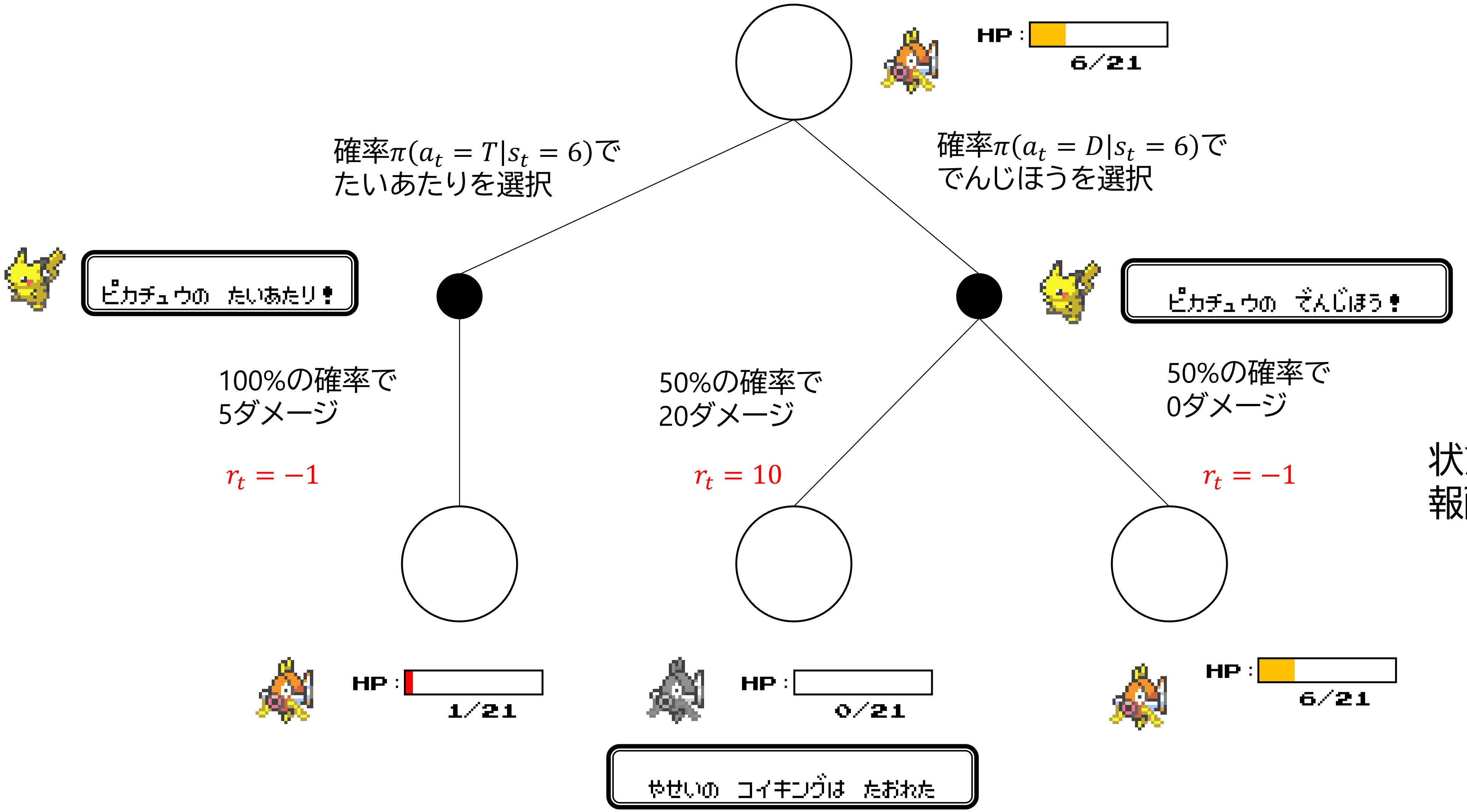
$$G_t=\sum_{i=1}^{t} r_i = r_1+r_2+\cdots+r_t$$

を求める. これが方策の良し悪しを判断する基準になる

報酬と強化学習

- 即時的な報酬ではなく, 累計報酬を基準とすることが強化学習の特徴
- 即時的な報酬だけを最大化するならば,
  ダメージ期待値の大きいでんじほうを常に選択すればよいだろう
- しかし, 実際にそのような戦略が最適ではないことは明らか

報酬が得られる様子



状態sの変化に応じて
報酬を受け取る

割引率

- 終端状態がないような問題を考えるとき，累計報酬は発散してしまう
- 仮に終端状態があったときにも，はじめのうちに大きな報酬を得る場合と，ものすごく時間が立ってから大きな報酬を得る場合とでは，前者のほうが良いはず
- このような問題に対処するため，**割引率**γを設定することがある

$$G_t = \sum_{i=1}^{\infty} \gamma^i r_i = r_1 + \gamma r_2 + \cdots + \gamma^n r_n + \cdots$$

状態価値関数

- **状態価値関数**$v_\pi(s)$は, 状態sから始めて, 方策πに従ったとき, その後全部の報酬の期待値を表す.

$$v_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s \right]$$

ベルマン方程式

- 状態価値関数$v_\pi(s_t)$は, 次の状態価値関数$v_\pi(s_{t+1})$を用いて以下のように表せる. これを**ベルマン方程式**と呼ぶ

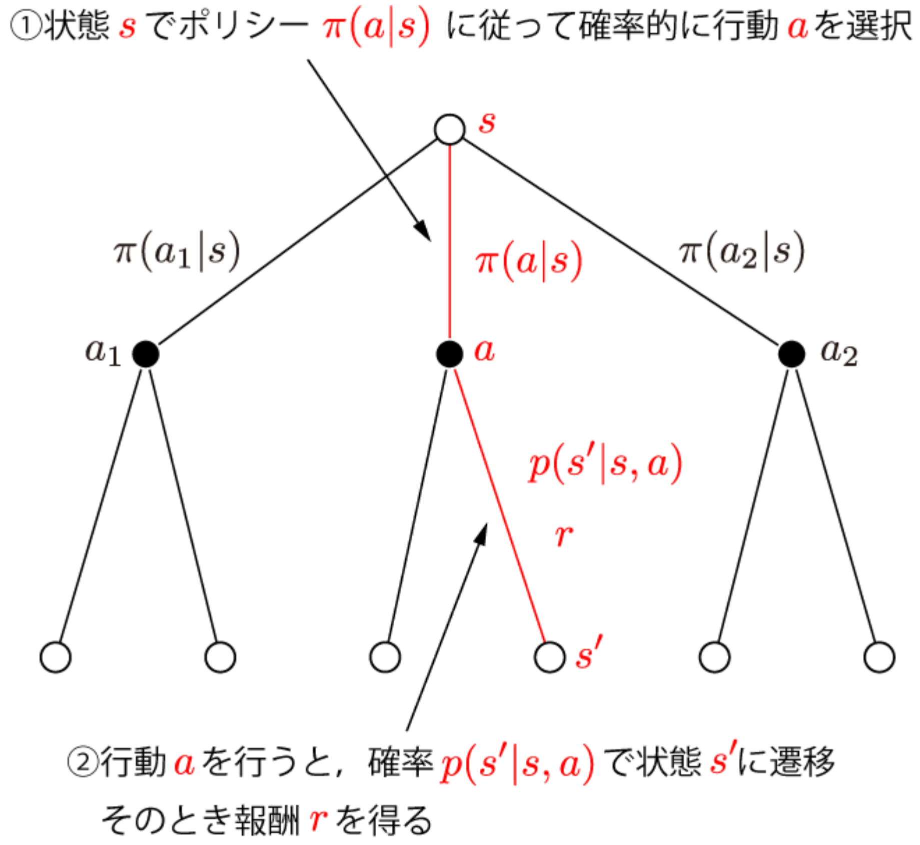
$$
\begin{aligned}
v_\pi(s) &= \mathbb{E}_\pi[G_t | S_t = s] \\
&= \mathbb{E}_\pi\left[R_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k R_{t+k+2} | S_t = s\right] \\
&= \sum_a \pi(a|s) \left\{ \sum_{s',r} p(s', r | s, a)\{r + \gamma v_\pi(s')\} \right\}
\end{aligned}
$$

aを行う確率

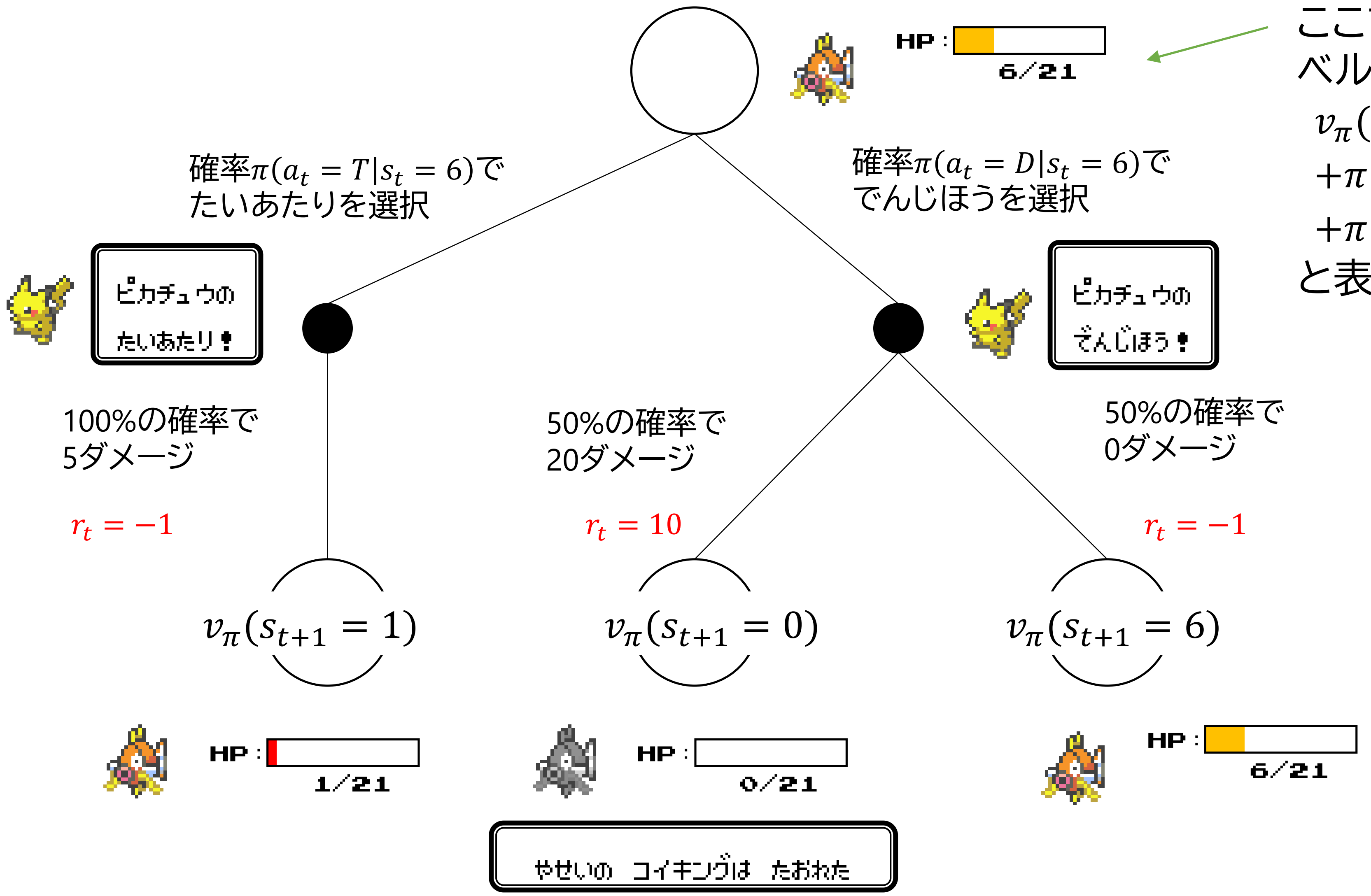
s'に遷移する確率

s'に遷移したことで得られる報酬

今後得られる報酬（割引含む）



ある場合のベルマン方程式



ここでの価値$v_\pi(s_t = 6)$は
ベルマン方程式にもとづき

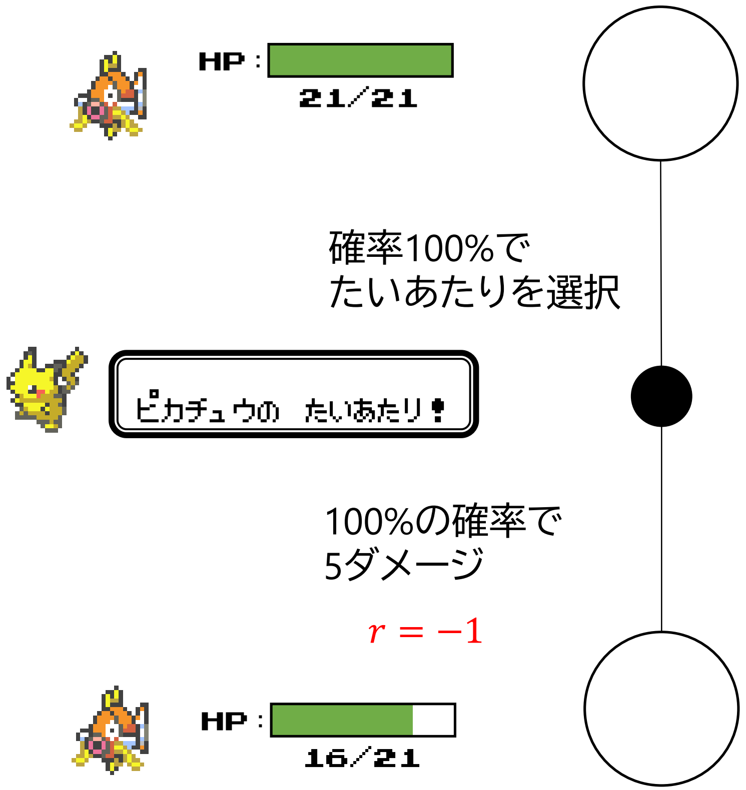
$$
\begin{aligned}
v_\pi(6) &= \pi(T|6) \times 1 \times \left(-1 + \gamma v_\pi(1)\right) \\
&+ \pi(D|6) \times 0.5 \times \left(10 + \gamma v_\pi(0)\right) \\
&+ \pi(D|6) \times 0.5 \times \left(-1 + \gamma v_\pi(6)\right)
\end{aligned}
$$

と表すことができる

ベルマン方程式を解いてみる①

簡単な場合には, ベルマン方程式を解析的に解くことができる

例1：$\pi(T) = 1, \pi(D) = 0$の場合. すなわち, たいあたりしかしない場合. $\gamma = 1$として割引なし.



まずはそれぞれの状態に対してベルマン方程式を書き下してみる

$$v_T(s_t = 21) = 1 \times 1 \times \left(-1 + v_T(s_{t+1} = 16)\right) = v_T(16) - 1$$

行動選択確率　　状態遷移確率

同様に

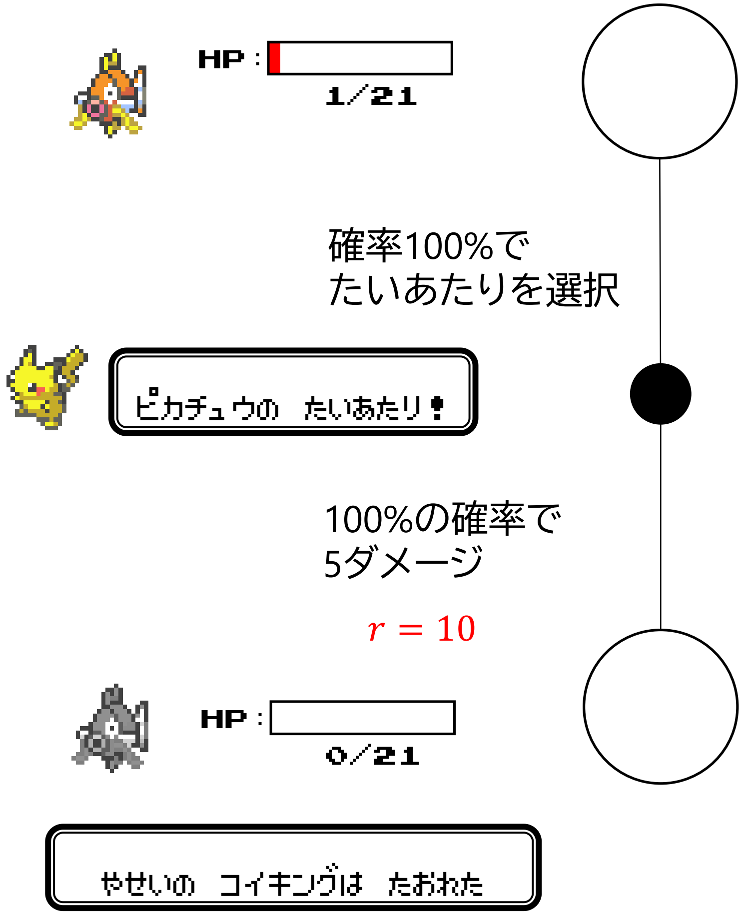
$$v_T(16) = v_T(11) - 1$$

$$v_T(11) = v_T(6) - 1$$

$$v_T(6) = v_T(1) - 1$$

ベルマン方程式を解いてみる①

例1：$\pi(T)=1, \pi(D)=0$の場合. すなわち,「たいあたり」しかしない場合. $\gamma=1$として割引なし.



終端状態では大きな報酬がもらえる

$$v_T(s_t=1)=1\times1\times\big(10+v_T(s_{t+1}=0)\big)=v_T(0)+10$$

終端状態に達した以降は報酬が発生しないので,

$$v_T(0)=0$$

あとは終端状態から逆に辿っていく

$$v_T(1)=v_T(0)+10=10$$

$$v_T(6)=v_T(1)-1=9$$

$$v_T(11)=v_T(6)-1=8$$

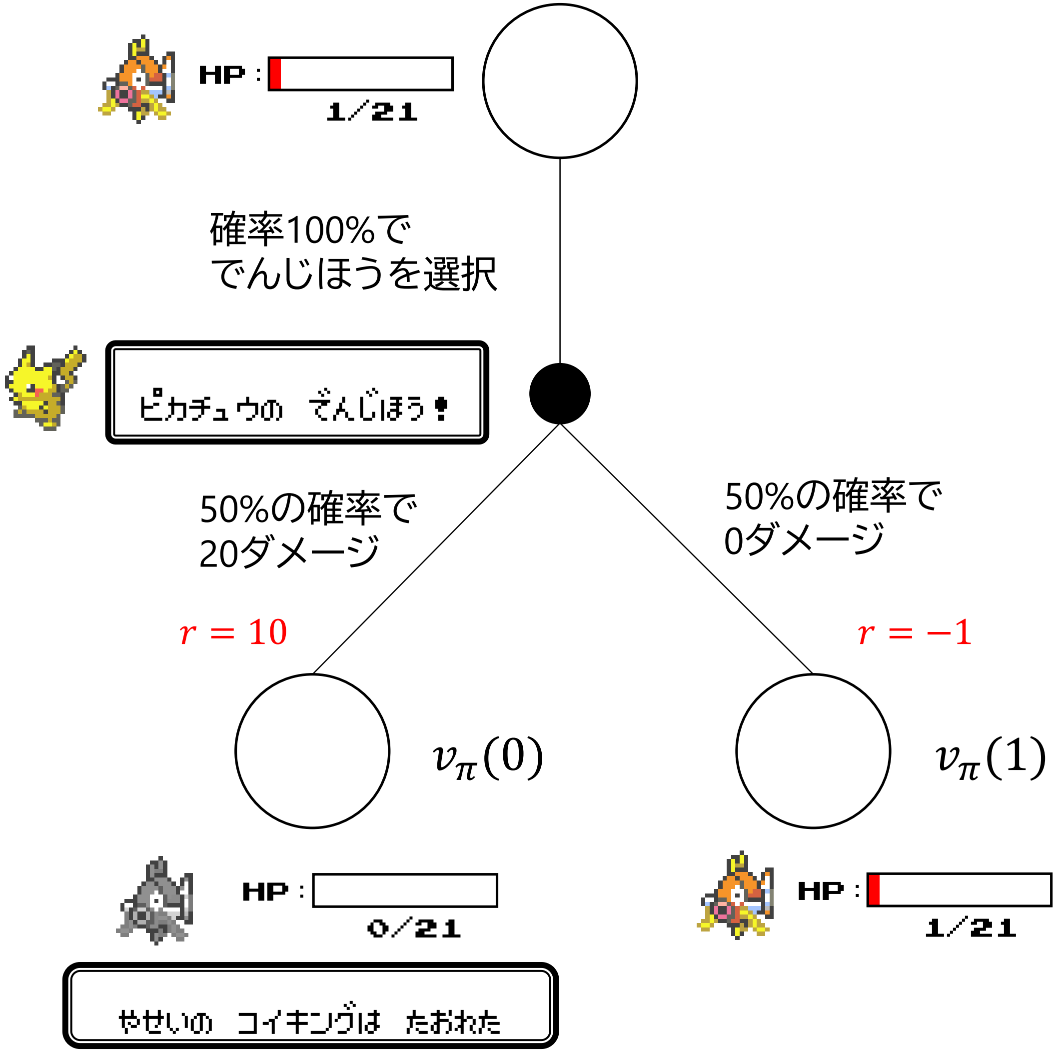
$$v_T(16)=v_T(11)-1=7$$

$$v_T(21)=v_T(16)-1=6$$

これで方策$\pi(T)=1$に対してすべての状態におけるベルマン方程式が解けた

ベルマン方程式を解いてみる②

例2：$\pi(T) = 0, \pi(D) = 1$の場合. すなわち,「でんじほう」しかしない場合. $\gamma = 1$として割引なし.



例1から, 終端状態から逆に辿っていくのがよい
終端状態では価値関数は

$$v_D(s_t = 0) = 0$$

次に, 残りHPが1のとき（左のバックアップ線図）

$$v_D(s_t = 1) = 1 \times 0.5 \times \left(10 + v_D(0)\right) + 1 \times 0.5 \times \left(-1 + v_D(1)\right)$$

これを整理して解くと

$$v_D(1) = 9$$

を得る

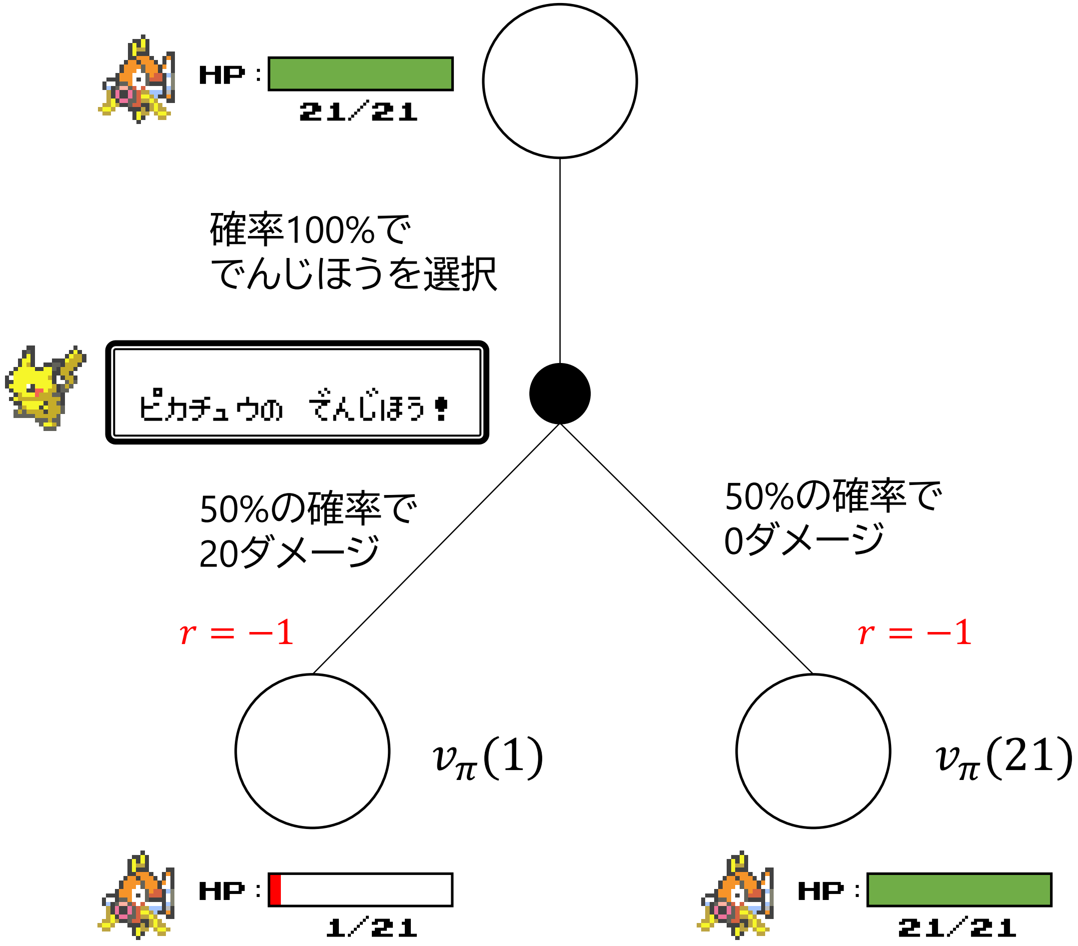
同様に

$$v_D(6) = v_D(11) = v_D(16) = 9$$

である

ベルマン方程式を解いてみる②

例2：$\pi(T) = 0, \pi(D) = 1$の場合. すなわち,「でんじほう」しかしない場合. $\gamma = 1$として割引なし.



HPが満タンから始まったとき（左のバックアップ線図）

$$v_D(s_t = 21) = 1 \times 0.5 \times \left(-1 + v_D(1)\right)$$
$$+1 \times 0.5 \times \left(-1 + v_D(21)\right)$$

これを整理して解くと

$$v_D(21) = 7$$

これで方策$\pi(D) = 1$に対して.
すべての状態におけるベルマン方程式が解けた.

状態価値関数の比較

- たいあたりしかしない場合と, でんじほうしかしない場合の状態価値関数を比較してみる

s	$v_T(s)$	$v_D(s)$
21	6	7
16	7	9
11	8	9
6	9	9
1	10	9
0	0	0

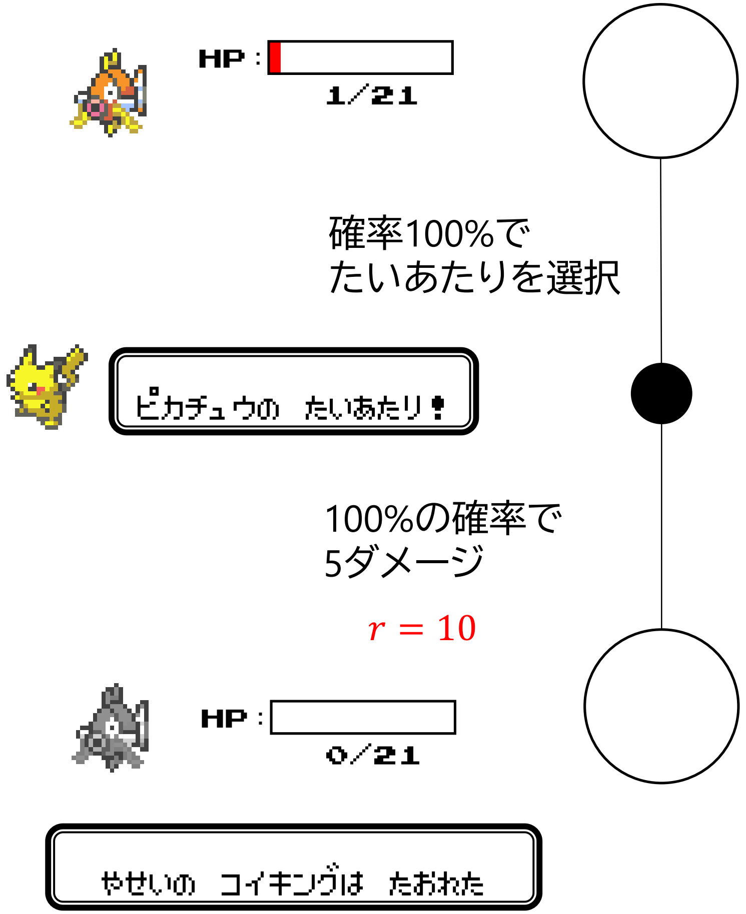
基本的には, でんじほうだけを選ぶ方策の方が価値関数が大きい

しかし, 残りHPが1のときには価値が逆転している
当然予想される通り, 「でんじほうだけを選ぶ」という方策は最適ではなさそうだ

以上の結果から,
「残りHPが1のときには必ずたいあたりを選択し, 他の場合は必ずでんじほうを選択」
という方策が最適(最も効率よくコイキングを倒せる)であることが予測できる

改善した方策π^*のベルマン方程式を解く

改善した方策：残りHPが1のときにはたいあたりを，他の場合はでんじほうを選択する



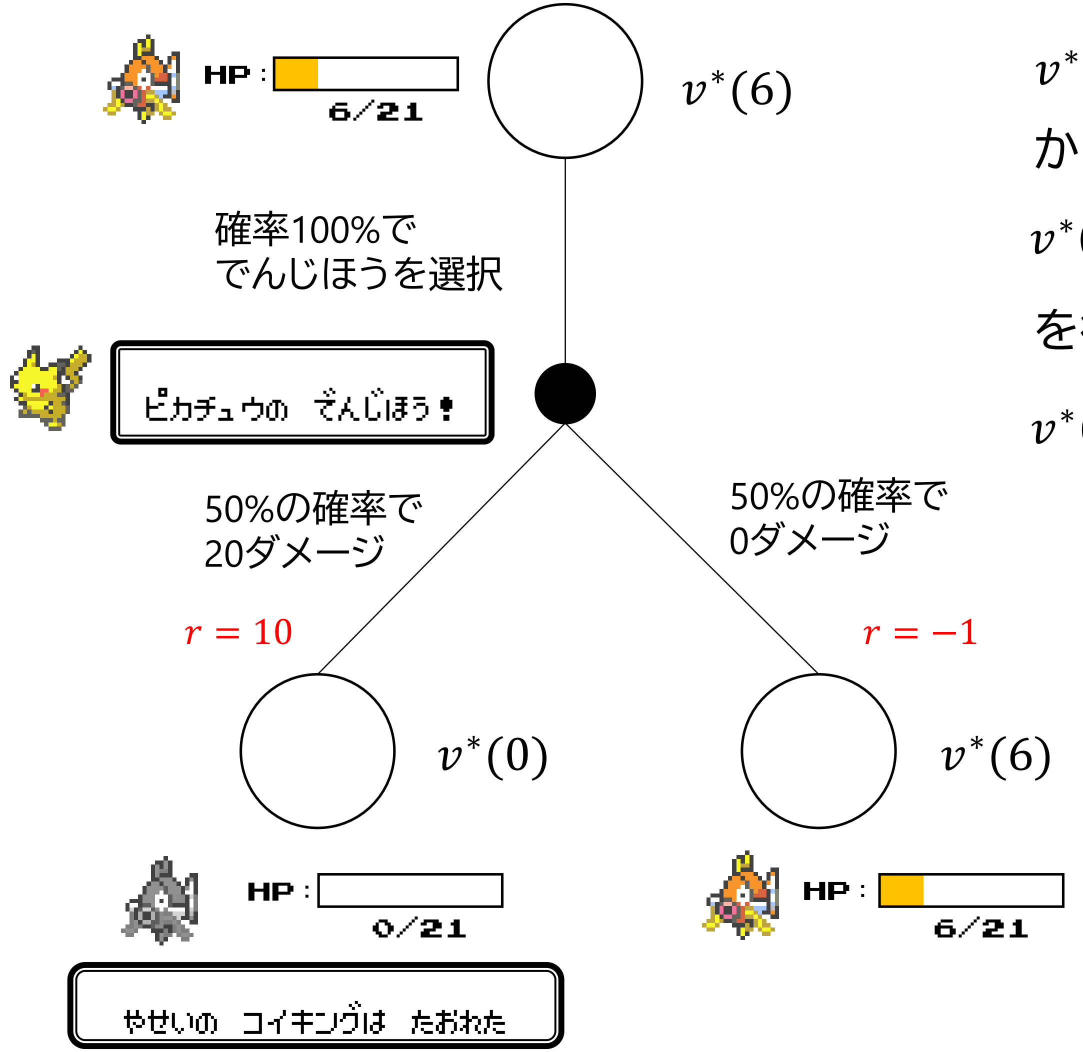
この方策に対して状態価値関数を求めていく

$$v^*(s_t = 0) = 0$$

$$v^*(s_t = 1) = 1 \times 1 \times \left(10 + v_T(0)\right) = v_T(0) + 10 = 10$$

改善した方策π^*のベルマン方程式を解く

改善した方策:残りHPが1のときにはたいあたりを, 他の場合はでんじほうを選択する



$$v^*(s_t = 6) = 1 \times 0.5 \times \left(10 + v^*(0)\right) + 1 \times 0.5 \times (-1 + v^*(6))$$

から

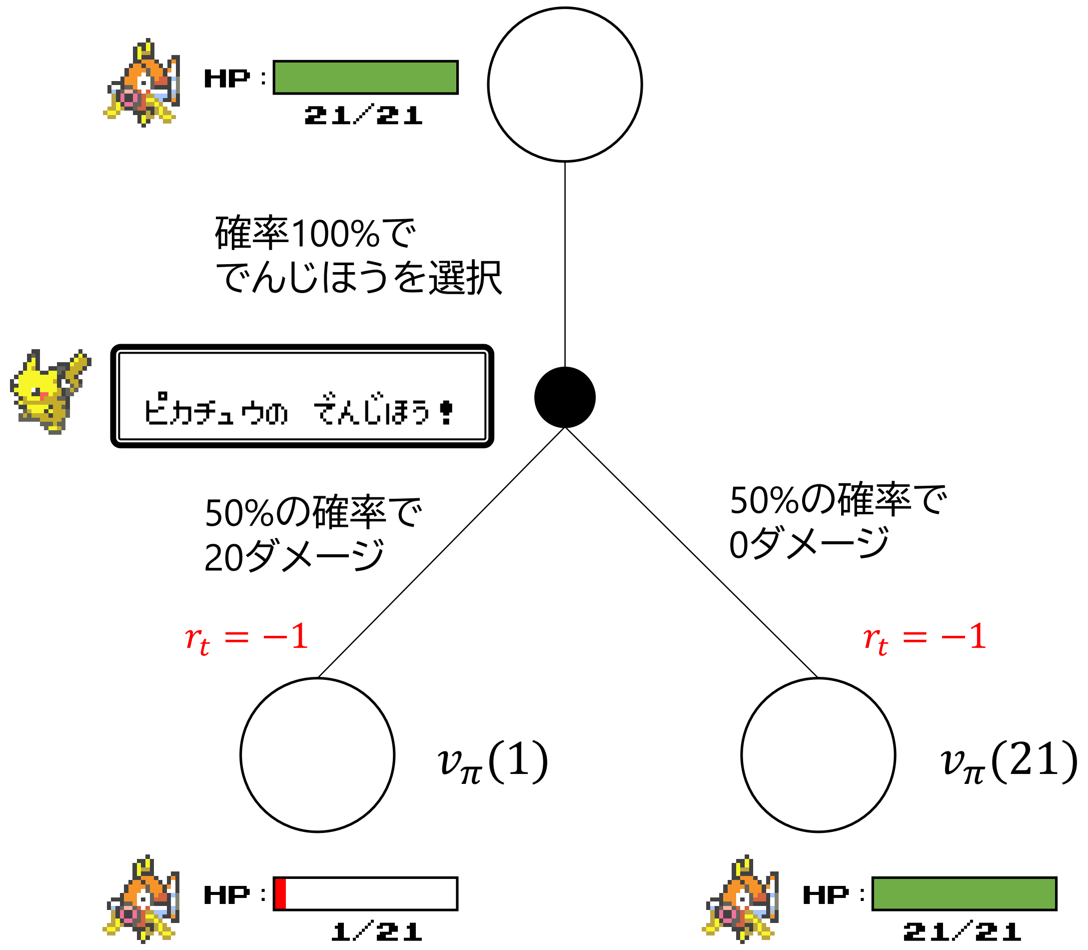
$$v^*(6) = 9$$

を得る. 同様にして

$$v^*(11) = v^*(16) = 9$$

改善した方策π^*のベルマン方程式を解く

最適方策:残りHPが1のときにはたいあたりを, 他の場合はでんじほうを選択する



HPが満タンから始まったとき(左のバックアップ線図)

$$v^*(s_t = 21) = 1 \times 0.5 \times \big(-1 + v^*(1)\big)$$
$$+1 \times 0.5 \times \big(-1 + v^*(21)\big)$$

これを整理して解くと

$$v_D(21) = 8$$

これで最適方策π^*に対して
すべての状態におけるベルマン方程式が解けた

状態価値関数の比較(改善した方策も含む)

- 改善した方策に対する状態価値関数が求められたので, 他の方策における価値関数と比較してみる

s	$v_T(s)$	$v_D(s)$	$v^*(s)$
21	6	7	8
16	7	9	9
11	8	9	9
6	9	9	9
1	10	9	10
0	0	0	0

すべての場合において, $v^*(s) \geq v_T(s)$と$v^*(s) \geq v_D(s)$が成立している
提案した方策は, 確かに他の方策よりも優れていることが証明できた

今後解決するべき課題:この方策は"最適"方策だろうか?他にもっと良い方策は存在する?

3. 最適方策を求める

行動価値関数

- どの行動を取るのが最適なのかを考えるための指標が欲しい
- 行動価値関数$q_\pi(s,a)$は, 状態sにおいて行動aを実行し, その後方策πに従ったとき, その後全部の報酬の期待値を表す.

$$q_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k+1}|S_t = s, A_t = a\right]$$

$$= \sum_{s',r} p(s',r|s,a)\left\{r + \gamma v_\pi(s')\right\}$$

状態価値関数のベルマン方程式と比較すると,
行動aが決定しているところが異なる

$$v_\pi(s) = \sum_a \pi(a|s)\left\{\sum_{s',r} p(s',r|s,a)\{r + \gamma v_\pi(s')\}\right\}$$

行動価値関数を求めてみよう

例:$\pi(T)=1, \pi(D)=0$の場合. すなわち,「たいあたり」しかしない場合. $\gamma=1$として割引なし.

例えば$s=16$に対して$a=T$を行う行動価値関数$q_T(16,T)$は,
$v_T(11)=8$ であることを用いて,

$$q_T(16,T)=1\times\left(-1+v_T(11)\right)=7$$

と求められる.
他の場合についても同様に求めると以下の表のようになる.

s	$q_T(s,T)$
21	6
16	7
11	8
6	9
1	10
0	0

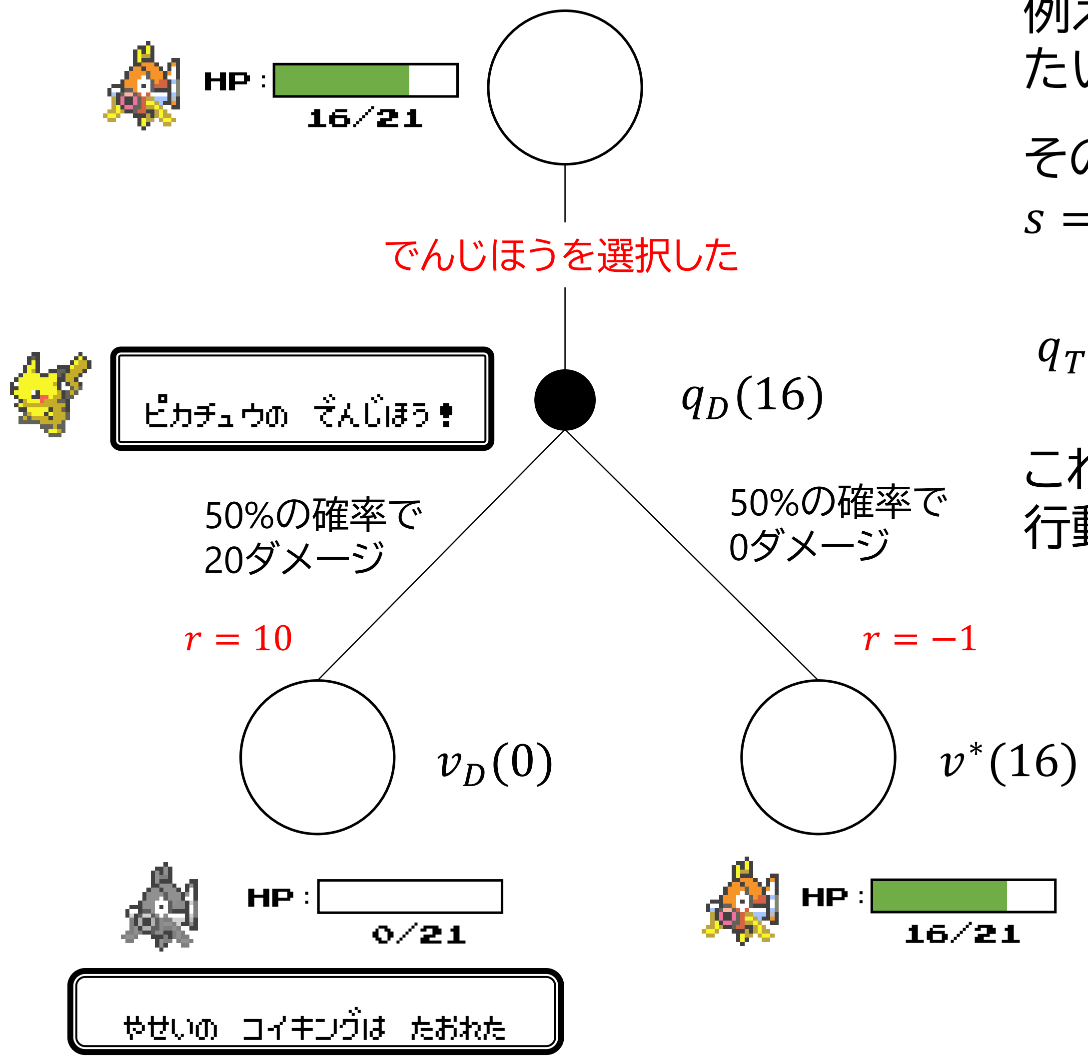
$v_T(s)$と変わらない

方策を改善するアイデア

- ある方策πが与えられたとき, ある状態sに対して, 取りうる全ての行動a（現在の方策的には選ばないものも）について行動価値関数$q(s,a)$を求める
- ある$a = a^*$を取ったときの行動価値関数$q(s,a^*)$が, もとの方策を取り続ける場合よりも大きい値を取る場合, 方策を変更したほうがよいと判断できる

行動価値関数を用いて，よりよい方法を探す

例：$\pi(T) = 1, \pi(D) = 0$の場合．すなわち，たいあたりしかしない場合．$\gamma = 1$として割引なし．



例えば$s = 16$の場合について，
たいあたりではなくでんじほうを選択する．

その他の場合には元通りたいあたりのみを選択するとき，
$s = 16, a = D$の行動価値関数は，$v_T(0) = 0, v_T(16) = 7$を用いて

$$q_{T'}(s = 16, a = D) = \frac{1}{2} \times \left(10 + v_T(0)\right) + \frac{1}{2} \times \left(-1 + v_T(16)\right) = 8$$

これは，この状況でたいあたりを選択するときの
行動価値関数$q_T(16, T) = 7$と比較して

$$q_{T'}(16, D) > q_T(16, T)$$

が成立する．すなわち，この方策を改善して
「たいあたり」よりも「でんじほう」を選択する方が
良いことが示された．

最適方策を探すには

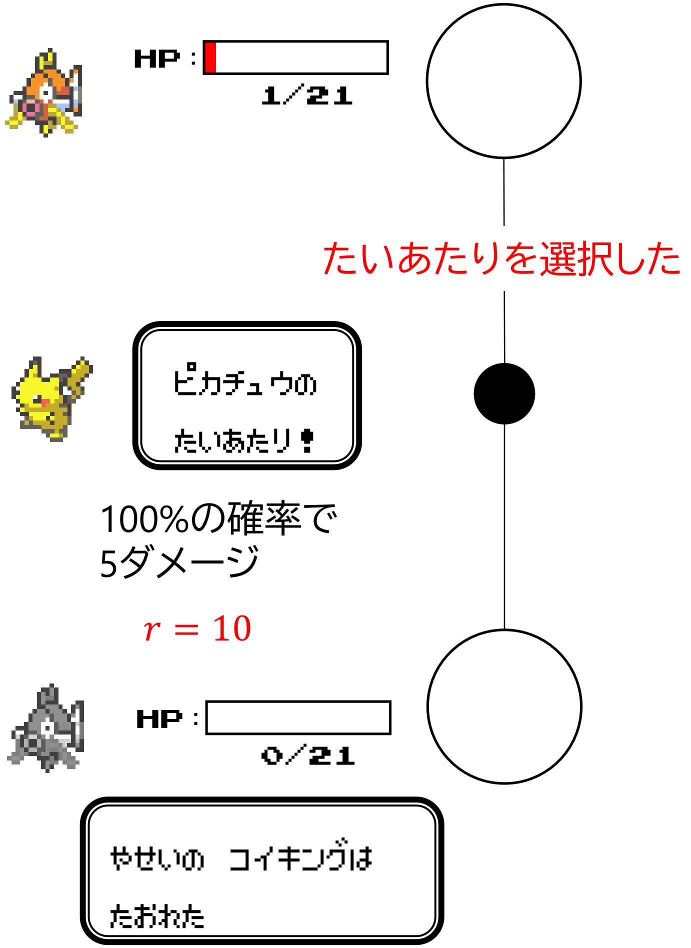
- ある適当な初期方策π_0を用意して, ある状態について, それよりも優れた(すなわち, 行動価値関数が大きくなるような)行動を選択するような, 更新方策π_1を見つける
- 方策π_1に対して, さらに優れた更新方策π_2を見つける. . . を繰り返すと, やがて最適方策π^*に収束することが理解できる
- このとき, 状態価値関数は最適価値関数v^*に収束する

$$v^* = \max_{a \in A(s)} q_{\pi^*}(s, a)$$

最適方策を求めてみよう(1/5)

- 初期方策π_0を,「でんじほう」だけを選択する方策とする

ここでも, 残りHPが1のときから始める



$s = 0$のとき, でんじほうを選択すると,

$$q(s = 1, a = D) = v_D(1) = 9$$

$s = 0$のとき, でんじほうではなく, たいあたりを選択すると,

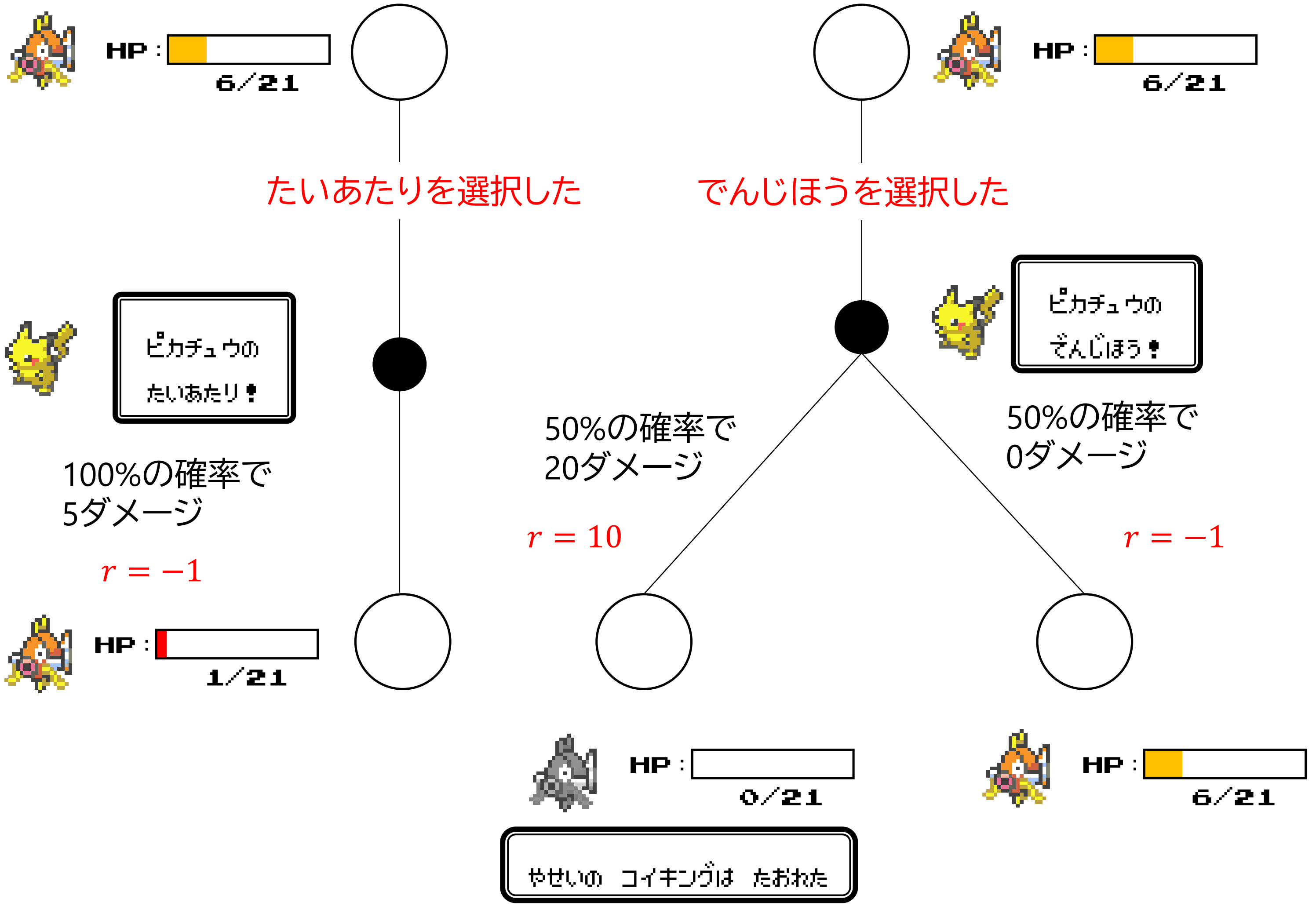
$$q(s = 1, a = T) = 1 \times \left(10 + v_D(0)\right) = 10$$

このとき, $q(1, D) < q(1, T)$が成立するので,
$s = 1$のときは, でんじほうではなく, たいあたりを選択したほうが良いことが示された.

改善した方策π_1：
「残りHPが1のときには, 必ずたいあたりを選択する.
他の場合には, 必ずでんじほうを選択する.」

最適方策を求めてみよう(2/5)

- 方策π_1をさらに更新していきたい



$s = 6$でたいあたりを選択

$$q(6, T) = 1 \times \left(-1 + v_{\pi_1}(1)\right) = 9$$

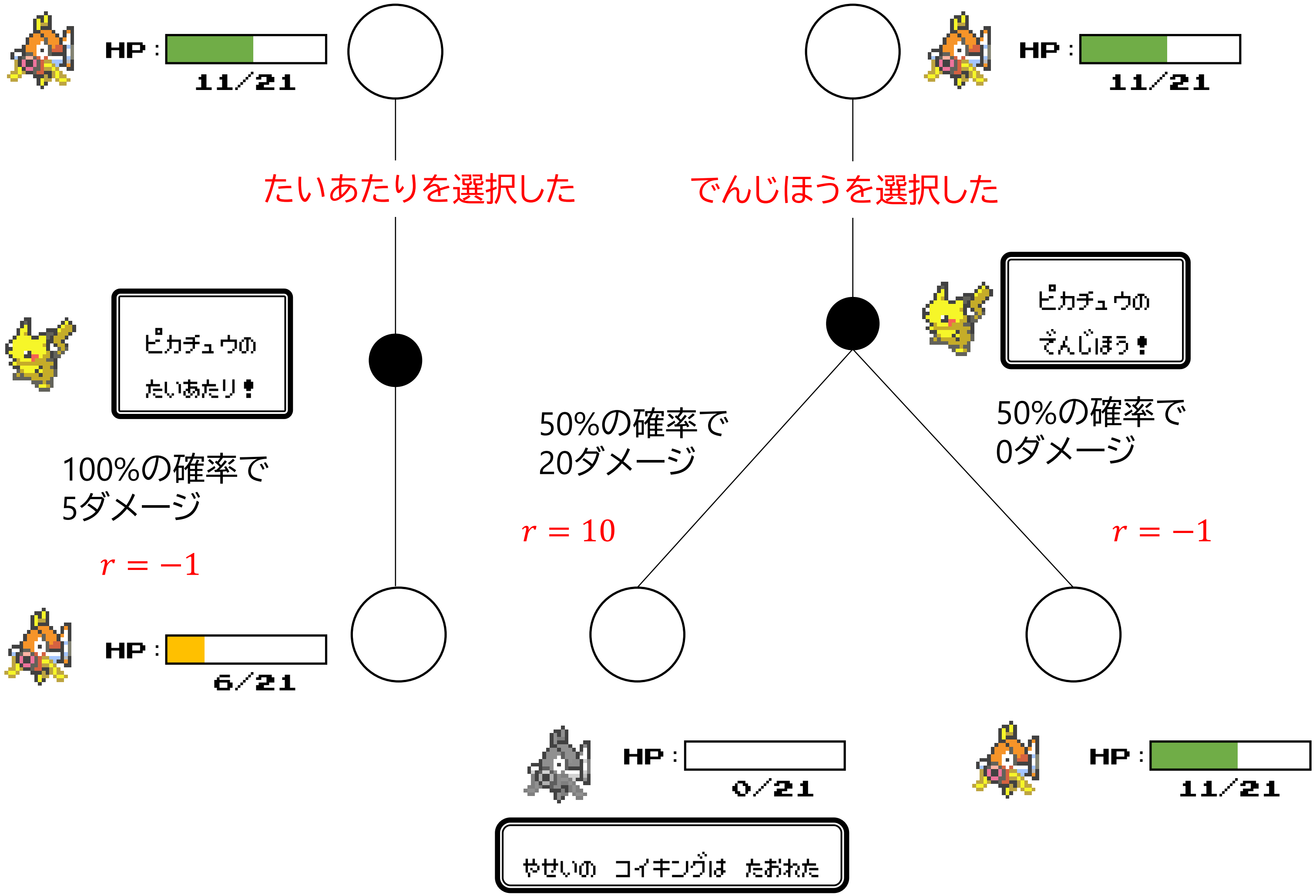
$s = 6$ででんじほうを選択

$$q(6, D) = 0.5 \times \left(10 + v_{\pi_1}(0)\right) + 0.5 \times (-1 + v_{\pi_1}(6)) = 9$$

ここでは, どちらの技を選択しても
行動価値関数は変化しない.
方策π_1をここで更新する必要はなさそうだ
($s = 6$ではでんじほうを選択)

最適方策を求めてみよう(3/5)

- 方策π_1をさらに更新していきたい



$s=11$でたいあたりを選択

$$q(11,T)=1\times\left(-1+v_{\pi_1}(6)\right)=8$$

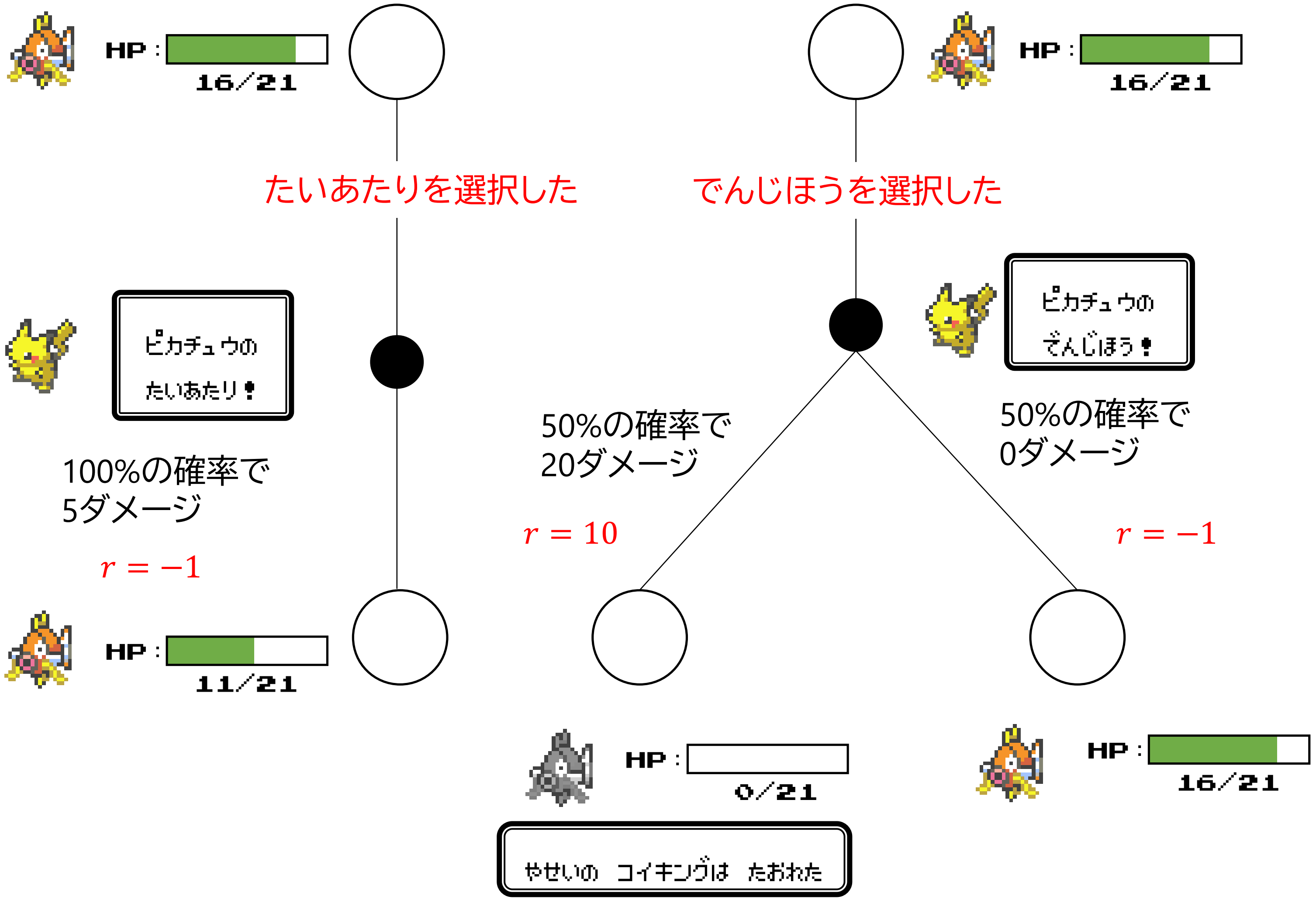
$s=11$ででんじほうを選択

$$q(11,D)=0.5\times\left(10+v_{\pi_1}(0)\right)$$
$$+0.5\times(-1+v_{\pi_1}(11))=9$$

ここでは方策π_1を更新する必要はない
($s=11$ではでんじほうを選択)

最適方策を求めてみよう(4/5)

- 方策π_1をさらに更新していきたい



$s = 16$でたいあたりを選択

$$q(16, T) = 1 \times \left(-1 + v_{\pi_1}(11)\right) = 8$$

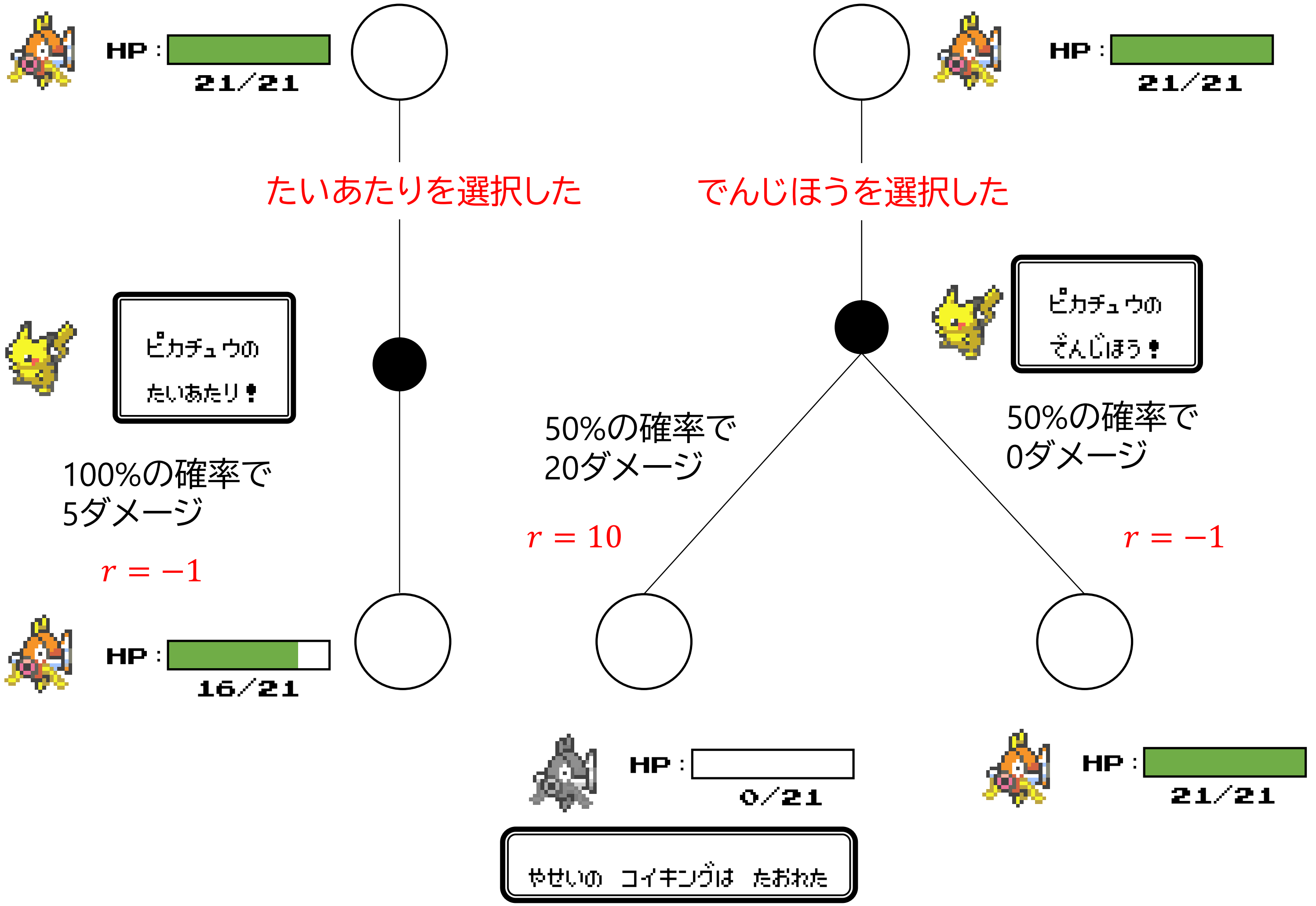
$s = 16$ででんじほうを選択

$$q(16, D) = 0.5 \times \left(10 + v_{\pi_1}(0)\right)$$
$$+0.5 \times (-1 + v_{\pi_1}(16)) = 9$$

ここでも方策π_1を更新する必要はない
（$s = 16$ではでんじほうを選択）

最適方策を求めてみよう(5/5)

- 方策π_1をさらに更新していきたい



$s = 21$でたいあたりを選択

$$q(21, T) = 1 \times \left(-1 + v_{\pi_1}(16)\right) = 8$$

$s = 21$ででんじほうを選択

$$q(21, D) = 0.5 \times \left(-1 + v_{\pi_1}(0)\right)$$
$$+0.5 \times (-1 + v_{\pi_1}(21)) = 8$$

ここでは方策π_1を更新してもよいし，
しなくてもよい
($s = 21$ではたいあたり，でんじほうの
いずれを選択してもよい)

状態価値関数の比較（改善した方策も含む）

- 方策の更新が終了し, 最適状態価値関数v^*が求められたので, 他の方策における価値関数と比較してみる

s	$v_T(s)$	$v_D(s)$	$v^*(s)$
21	6	7	8
16	7	9	9
11	8	9	9
6	9	9	9
1	10	9	10
0	0	0	0

確かに$v^*(s)$は最適方策π_1に対する最適価値関数になっていることが示された

まとめ

- 強化学習は, 環境の状態に対して, エージェントが取る行動を決定するための方策を最適化する
- 方策は, 即時的な報酬ではなく, その累計の期待値である価値関数を最大化するように決定する
- ベルマン方程式は, 現在の状態における状態価値関数と次の状態における状態価値関数の関係を表す
- 方策を改善するためには, ある行動を起こしてみたときの行動価値関数を比較する