A Markov Game Model for Valuing Player Actions in Ice Hockey

Kurt Routley
School of Computing Science
Simon Fraser University
Vancouver, BC, Canada
kdr4@sfu.ca

Oliver Schulte
School of Computing Science
Simon Fraser University
Vancouver, BC, Canada
oschulte@cs.sfu.ca

Abstract

A variety of advanced statistics are used to evaluate player actions in the National Hockey League, but they fail to account for the context in which an action occurs or to look ahead to the long-term effects of an action. We apply the Markov Game formalism to develop a novel approach to valuing player actions in ice hockey that incorporates context and lookahead. Dynamic programming is used to learn Q-functions that quantify the impact of actions on goal scoring resp. penalties. Learning is based on a massive dataset that contains over 2.8M events in the National Hockey League. The impact of player actions is found to vary widely depending on the context, with possible positive and negative effects for the same action. We show that lookahead makes a substantial difference to the action impact scores. Players are ranked according to the aggregate impact of their actions. We compare this impact ranking with previous player metrics, such as plus-minus and salary.

1 INTRODUCTION

A fundamental goal of sports statistics is to understand which actions contribute to winning in what situation. As sports have entered the world of big data, there is increasing opportunity for large-scale machine learning to model complex sports dynamics. The research described in this paper applies AI techniques to model the dynamics of ice hockey; specifically the Markov Game model formalism [Littman, 1994], and related computational techniques such as the dynamic programming value iteration algorithm. We make use of a massive dataset about matches in the National Hockey League (NHL). This dataset comprises all play-by-play events from 2007 to 2014, for a total of over 2.8M events/actions and almost 600K play sequences. The Markov Game model comprises over 1.3M states. Whereas most previous works on Markov Game models aim to compute optimal strategies or policies [Littman, 1994] (i.e., minimax or equilibrium strategies), we learn a model of how hockey is actually played, and do not aim to compute optimal strategies. In reinforcement learning (RL) terminology, we use dynamic programming to compute an *action-value* Q-function in the *"on policy"* setting [Sutton and Barto, 1998]. In RL notation, the expression $Q(s, a)$ denotes the expected reward of taking action a in state s.

Motivation Motivation for learning a Q-function for NHL hockey dynamics includes the following.

Knowledge Discovery. The Markov Game model provides information about the likely consequences of actions. The basic model and algorithms can easily be adapted to study different outcomes of interest, such as goals and penalties.

Player Evaluation. One of the main tasks for sports statistics is evaluating the performance of players [Schumaker et al., 2010]. A common approach is to assign action values, and sum the corresponding values each time a player takes the respective action. An advantage of this additive approach is that it provides highly interpretable player rankings. A simple and widely used example in ice hockey is the +/- score: for each goal scored by (against) a player's team when he is on the ice, add +1 (-1) point. Researchers have developed several extensions of +/- for hockey [Macdonald, 2011; Spagnola, 2013; Schuckers and Curro, 2013].

There are two major problems with the previous action count approaches used in ice hockey. (1) They are unaware of the *context* of actions within a game. For example, a goal is more valuable in a tied-game situation than when the scorer's team is already four goals ahead [Pettigrew, 2015]. Another example is that if a team manages two successive shots on goal, the second attempt typically has a higher chance of success. In the Markov Game model, *context = state.* Formally, the Q function depends *both* on the state s and the action a. Richer state spaces therefore capture more of the context of an action. (2) Previous action scores are based on immediate positive consequences

of an action (e.g. goals following a shot). However, an action may have medium-term and/or ripple effects rather than immediate consequences in terms of visible rewards like goals. Therefore evaluating the impact of an action requires *lookahead*. Long-term lookahead is especially important in ice hockey because evident rewards like goals occur infrequently [Lock and Schuckers, 2009]. For example, if a player receives a penalty, this leads to a manpower disadvantage for his team, known as a power play for the other team. It is easier to score a goal during a power play, but this does not mean that a goal will be scored immediately after the penalty. For another example, if a team loses the puck in their offensive zone, the resulting counterattack by the other team may lead to a goal eventually but not immediately. The dynamic programming value iteration algorithm of Markov Decision Processes provides a computationally efficient way to perform unbounded lookahead.

Evaluation Our evaluation learns Q-functions for two reward events, scoring the next goal and receiving the next penalty. We observe a wide variance of the impact of actions with respect to states, showing context makes a substantial difference. We provide examples of the context dependence to give a qualitative sense of how the Markov Game model accounts for context. To evaluate player performance, we use the Q-function to quantify the value of a player's action in a context. The action values are then aggregated over games and seasons to get player impact scores. Player impact scores correlate with plausible alternative scores, such as a player's total points, but improve on these measures, as our impact score is based on many more events.

Contributions We make our extensive dataset available on-line, in addition to our code and the learned Markov game model [Routley et al., 2015]. The main contributions of this paper may be summarized as follows:

1. The first Markov Game model for a large ice hockey state space (over 1.3M), based on play sequence data.

2. Learning a Q-function that models play dynamics in the National Hockey League from a massive data set (2.8M events). We introduce a variant of AD-Trees as a data structure to (1) compute and store the large number of sufficient statistics required [Moore and Lee, 1998], and (2) support value iteration updates.

3. Applying the Q-function to define a context-aware look-ahead measure of the value of an action, over configurable objective functions (rewards).

4. Applying the context-aware action values to score hockey player actions, including how players affect penalties as well as goals.

Paper Organization. We review related work in measuring player contributions and machine learning in sports in Section 2. We then give some background information on the ice hockey domain and NHL play-by-play sequences data. Our Markov Game model translates the hockey domain features into the Markov formalism. We then discuss how we implement scalable value iteration for the ice hockey domain. The evaluation section addresses the impact of context and lookahead. We apply the model to rank the aggregate performance of players and describe the resulting player ranking. We view our work as taking the first step, not the last, in applying AI modelling techniques to ice hockey. Therefore we conclude with a number of potential extensions and open problems for future work.

2 RELATED WORK

Markov Process Models for Ice Hockey A number of Markov process models have been developed for ice hockey [Thomas et al., 2013; Buttrey et al., 2011]. The main difference to our work is these models do not include actions, and hence cannot model the impact of actions.

Markov Decision Process Models for Other Sports MDP-type models have been applied in a number of sports settings, such as baseball [Sidhu and Caffo, 2014] and soccer [Hirotsu et al., 2002].Our work is similar in that our method uses value iteration on a Markovian state space, however, previous Markov models in sports use a much smaller state space. For example, the baseball model of [Sidhu and Caffo, 2014] utilizes only 12 states compared to the 1,325,809 states in our model. The goal of these models is to find an optimal policy for a critical situation in a sport or game. In contrast, we learn in the on-policy setting whose aim is to model hockey dynamics as it is actually played.

Evaluating Actions and Players in Ice Hockey Several papers aim to improve the basic +/- score with statistical techniques [Macdonald, 2011; Gramacy et al., 2013; Spagnola, 2013]. A common approach is to use regression techniques where an indicator variable for each player is used as a regressor for a goal-related quantity (e.g., log-odds of a goal for the player's team vs. the opposing team). The regression weight measures the extent to which the presence of a player contributes to goals for his team or prevents goals for the other team. These approaches look at only goals, no other actions. The only context they take into account is which players are on the ice when a goal is scored.

The closest predecessor to our work in ice hockey is the Total Hockey Rating (THoR) [Schuckers and Curro, 2013]. This assigns a value to all actions, not only goals. Actions were evaluated based on whether or not a goal occurred in the following 20 seconds after an action. This used data

from the 2006/2007 NHL season only. THoR assumes a fixed value for every action and does not account for the context in which an action takes place. Furthermore, the window of 20 seconds restricts the lookahead value of each action. Our Q-learning method is not restricted to any particular time window for lookahead.

Evaluating Actions and Players in Other Sports [Cervone et al., 2014] uses spatial-temporal tracking data for basketball to build the POINTWISE model for valuing player decisions and player actions. Conceptually, their approach to defining action values is the closest predecessor to ours: The counterpart to the value of a state in a Markov game is called expected possession value (EPV). The counterpart to the impact of an action on this value is called EPV-added (EPVA). Cervone *et al.* emphasize the broad potential of the context-based impact definitions: "we assert that most questions that coaches, players, and fans have about basketball, particularly those that involve the offense, can be phrased and answered in terms of EPV."

While the definition of action impact is conceptually very similar, [Cervone et al., 2014] uses neither AI terminology nor AI techniques, which we cover in this paper. Moreover, all the underlying details are different between our model and theirs: [Cervone et al., 2014] discuss the advantages of using a discrete state space for stochastic consistency, but consider it computationally infeasible for their data. We show that leveraging AI data structures and algorithms makes handling a large discrete state space feasible for ice hockey. Including the local action history in the state space allows us to capture the medium-term effects of actions. This is more important for ice hockey than for basketball, because scoring in basketball occurs at much shorter intervals.

3 DOMAIN DESCRIPTION: HOCKEY RULES AND HOCKEY DATA

We outline the rules of hockey and describe the dataset available from the NHL.

3.1 HOCKEY RULES

We describe a Markov Game Model for ice hockey. To motivate our model, we give a brief overview of rules of play in the NHL [National Hockey League, 2014]. NHL games consist of three periods, each 20 minutes in duration. A team will try to score more goals than their opponent within three periods in order to win the game. If the game is still tied after three periods, the teams will enter a fourth overtime period, where the first team to score a goal wins the game. If the game is still tied after overtime during the regular season, a shootout will commence. During the playoffs, overtime periods are repeated until a team scores a goal to win the game. Teams have five skaters and one goalie on the ice during even strength situations. Penalties result in a player sitting in the penalty box for two, four, or five minutes and the penalized team will be shorthanded, creating a manpower differential between the two teams. The period where one team is penalized is called a powerplay for the opposing team with a manpower advantage. A shorthanded goal is a goal scored by the penalized team, and a powerplay goal is a goal scored by the team on the powerplay.

3.2 DATA FORMAT

The NHL provides information about sequences of play-by-play events, which are scraped from `http://www.nhl.com` and stored in a relational database. The real-world dataset is formed from $2,827,467$ play-by-play events recorded by the NHL for the complete 2007-2014 seasons, regular season and playoff games, and the first 512 games of the 2014-2015 regular season. A breakdown of this dataset is shown in Table 1. The type of events recorded by the NHL from the 2007-2008 regular season and onwards are listed in Table 2. There are two types of events: actions performed by players and start and end markers for each play sequence. Every event is marked with a continuous timestamp, and every action is also marked with a zone Z and which team, Home or Away, carries out the action.

Table 1: Size of Dataset

Number of Teams	32
Number of Players	1,951
Number of Games	9,220
Number of Sequences	590,924
Number of Events	2,827,467

Table 2: NHL Play-By-Play Events Recorded

Action Event	**Start/End Event**
Faceoff	Period Start
Shot	Period End
Missed Shot	Early Intermission Start
Blocked Shot	Penalty
Takeaway	Stoppage
Giveaway	Shootout Completed
Hit	Game End
Goal	Game Off
	Early Intermission End

4 MARKOV GAMES

In its general form, a Markov Game [Littman, 1994], sometimes called a stochastic game, is defined by a set of states, S, and a collection of action sets, one for each agent in the environment. State transitions are controlled by the current state and one action from each agent. For each agent, there is an associated reward function mapping a state transition to a reward. An overview of how our Markov Game model fills in this schema is as follows. There are two players, the Home Team H and the Away Team A. In each state, only one team performs an action, although not in a turn-based sequence. This reflects the way the NHL records actions. Thus at each state of the Markov Game, exactly one player chooses No-operation. State transitions follow a semi-episodic model [Sutton and Barto, 1998] where play moves from episode to episode, and information from past episodes is recorded as a list of *context features*. The past information includes the goal score and manpower. A sequence in the NHL play-by-play data corresponds to an episode in Markov decision process terminology. *Within* each episode/sequence, our game model corresponds to a game tree with perfect information as used in AI game research [Russell and Norvig, 2010]. We introduce the following generic notation for all states. MDP notation follows [Russell and Norvig, 2010; Littman, 1994].

- $Occ(s)$ is the number of occurrences of state s as observed in the play-by-play data.
- $Occ(s, s')$ is the number of occurrences of state s being immediately followed by state s' as observed in the play-by-play data. (s, s') forms an edge in the transition graph of the Markov Game model.
- The transition probability function TP is a mapping of $S \times S \rightarrow (0, 1]$. We estimate it using the observed transition frequency $\frac{Occ(s,s')}{Occ(s)}$.

We begin by defining context features, then play sequences.

4.1 STATE SPACE: CONTEXT FEATURES

Previous work on Markov process models for ice hockey [Thomas et al., 2013] defined states in terms of hand-selected features that are intuitively relevant for the game dynamics, such as the goal differential and penalties. We refer to such features as **context features**. Context features remain the same throughout each play sequence.

A **context state** lists the values of relevant features at a point in the game. These features are shown in Table 3, together with the range of integer values observed.

Table 3: Context Features

Notation	Name	Range
GD	Goal Differential	[-8,+8]
MD	Manpower Differential	[-3,3]
P	Period	[1,7]

Goal Differential GD is calculated as Number of Home Goals - Number of Away Goals. A positive (negative) goal differential means the home team is leading (trailing). Manpower Differential MD is calculated as Number of Home Skaters on Ice - Number of Away Skaters on Ice. A positive manpower differential typically means the home team is on the powerplay (away team is penalized), and a negative manpower differential typically means the home team is shorthanded (away team is on the powerplay).[1] Period P represents the current period number the play sequence occurs in, typically ranging in value from 1 to 5. Periods 1 to 3 are the regular play of an ice hockey game, and periods 4 and onwards are for overtime and shootout periods as needed.

Potentially, there are $(17 \times 7 \times 7) = 833$ context states. In our NHL dataset, 450 context states occur at least once. Table 4 includes statistics for the top-20 context states over all $590,924$ play sequences, and lists $52,793$ total goals and $89,612$ total penalties. Positive differences are for the home team and negative differences are for the away team. For example, a Goal Difference of 7.1% means the home team is 7.1% more likely to score a goal in that context state than the away team. Similarly, a Penalty Difference of -33.2% means the away team is 33.2% more likely to receive a penalty in that context state than the home team. Our model is very well calibrated, meaning that its predictions match the observed frequencies of goals and penalties. We explain below how the model predictions are computed.

A number of previous papers on hockey dynamics have considered the context features of play sequences. The important trends that it is possible to glean from statistics such as those shown in Table 4 have been discussed in several papers. Our data analysis confirms these observations on a larger dataset than previously used. Notable findings include the following.

1. Home team advantage: the same advantages in terms of context features translate into higher scoring rates.
2. Penalties are more frequent than goals, except for the overtime period 4 (cf. [Schuckers and Brozowski, 2012]).
3. Gaining a powerplay substantially increases the conditional probability of scoring a goal [Thomas et al.,

[1] Pulling the goalie can also result in a skater manpower advantage.

Table 4: Statistics for Top-20 Most Frequent Context States. GD = Goal Differential, MD = Manpower Differential, P = Period.

						Observed		Model Predicts	
GD	**MD**	**P**	**#Sequences**	**#Goals**	**#Penalties**	**Goal Difference**	**Penalty Difference**	**Goal Difference**	**Penalty Difference**
0	0	1	78,118	5,524	11,398	7.06%	-2.26%	7.06%	-2.26%
0	0	2	38,315	2,935	5,968	7.60%	-2.92%	7.60%	-2.92%
0	0	3	30,142	2,050	3,149	5.85%	-2.19%	5.85%	-2.19%
1	0	2	29,662	2,329	4,749	2.02%	2.17%	2.02%	2.17%
1	0	3	25,780	2,076	3,025	4.34%	3.54%	4.34%	3.54%
-1	0	2	25,498	1,970	4,044	8.63%	-8.70%	8.63%	-8.70%
1	0	1	24,721	1,656	4,061	5.31%	3.42%	5.31%	3.42%
-1	0	3	22,535	1,751	2,565	0.74%	-18.28%	0.74%	-18.28%
-1	0	1	20,813	1,444	3,352	4.57%	-8.05%	4.57%	-8.05%
2	0	3	17,551	1,459	2,286	6.92%	-0.87%	6.92%	-0.87%
2	0	2	15,419	1,217	2,620	2.71%	2.90%	2.71%	2.90%
-2	0	3	13,834	1,077	1,686	-2.32%	-12.57%	-2.31%	-12.57%
0	1	1	12,435	1,442	2,006	64.77%	31.70%	64.77%	31.70%
-2	0	2	11,799	882	1,927	3.85%	-15.72%	3.85%	-15.72%
0	-1	1	11,717	1,260	2,177	-54.76%	-44.79%	-54.76%	-44.79%
3	0	3	10,819	678	1,859	0.29%	1.24%	0.29%	1.24%
-3	0	3	7,569	469	1,184	7.04%	-6.25%	7.04%	-6.25%
0	1	2	7,480	851	1,157	56.99%	25.67%	56.99%	25.67%
0	0	4	7,024	721	535	5.69%	-10.65%	5.69%	-10.65%
0	-1	2	6,853	791	1,150	-52.47%	-37.39%	-52.47%	-37.39%

2013].

4. Gaining a powerplay also significantly increases the conditional probability of receiving a penalty [Schuckers and Brozowski, 2012].

5. Shorthanded goals are surprisingly likely: a manpower advantage translates only into a goal scoring difference of at most 64.8%, meaning the shorthanded team scores the next goal with a conditional probability of 17.6%. (Powerplay for the home team in period 1.)

While such patterns provide interesting and useful insights into hockey dynamics, they do not consider action events. This means that analysis at the sequence level does not consider the internal dynamics within each sequence, and that it is not suitable for evaluating the impact of hockey actions. We next extend our state space to include actions.

4.2 STATE SPACE: PLAY SEQUENCES

We expand our state space with actions and action histories. The basic set of 8 possible actions is listed in Table 2. Each of these actions has two parameters: which team performs the action and the zone Z where the action takes place. Zone Z represents the area of the ice rink in which an action takes place. Z can have values Offensive, Neutral, or Defensive, relative to the team performing an action. For example, $Z =$ Offensive zone relative to the home team is equivalent to $Z =$ Defensive zone relative to the away team. A specification of an action plus parameters is an **action event**. Using action description language notation [Levesque et al., 1998], we write action events in the form $a(T, Z)$. For example, $faceoff(Home, Neutral)$ denotes the home team wins a faceoff in the neutral zone. We usually omit the action parameters from generic notation and write a for a generic action event.

A **play sequence** h is a sequence of events starting with exactly one start marker, followed by a list of action events, and ended by at most one end marker. Start and end markers are shown in Table 2, adding shots and faceoffs as start markers, and goals as end markers. We also allow empty history $\emptyset$ as a valid play sequence. A **complete** play sequence ends with an end marker. A **state** is a pair $s = \langle \mathbf{x}, h \rangle$ where $\mathbf{x}$ denotes a list of context features and h an action history. State s represents a play sequence consisting of action events $a_1, a_2, \ldots, a_n$ and with a particular GD, MD, and P as the context. If the sequence h is empty, then state s is purely a context node. Table 5 shows an example of a NHL play-by-play action sequence in tabular form. Potentially, there are $(7 \times 2 \times 3)^{40} = 42^{40}$ action histories. In our dataset, 1,325,809 states, that is, combinations of context features and action histories, occur at least once. We store sequence data in SQL tables (see Table 5). SQL provides fast retrieval, and native support for the necessary COUNT operations.

Table 5: Sample Play-By-Play Data in Tabular Format

GameId	Period	Sequence Number	Event Number	Event
1	1	1	1	PERIOD START
1	1	1	2	faceoff(Home,Neutral)
1	1	1	3	hit(Away,Neutral)
1	1	1	4	takeaway(Home,Defensive)
1	1	1	5	missed_shot(Away,Offensive)
1	1	1	6	shot(Away,Offensive)
1	1	1	7	giveaway(Away,Defensive)
1	1	1	8	takeaway(Home,Offensive)
1	1	1	9	missed_shot(Away,Offensive)
1	1	1	10	goal(Home,Offensive)
1	1	2	11	faceoff(Away,Neutral)
...				

4.3 STATE TRANSITIONS

If h is an incomplete play sequence, we write $h \star a$ for the play sequence that results from appending a to h, where a is an action event or an end marker. Similarly if $s = \langle \mathbf{x}, h \rangle$, then $s \star a \equiv \langle \mathbf{x}, h \star a \rangle$ denotes the unique successor state that results from executing action a in s. This notation utilizes the fact that context features do not change until an end marker is reached. For example, the goal differential does not change unless a goal event occurs. If h is a complete play sequence, then the state $\langle \mathbf{x}, h \rangle$ has a unique successor $\langle \mathbf{x}', \emptyset \rangle$, where the mapping from $\mathbf{x}$ to $\mathbf{x}'$ is determined by the end marker. For instance, if the end marker is $goal(Home, *)$, then the goal differential increases by 1. A sample of our state transition graph is shown in Figure 1. Note that $R(s)$ is the reward value for the state, and will be discussed in Section 4.4. In Figure 1, the reward encodes the objective of scoring a goal.

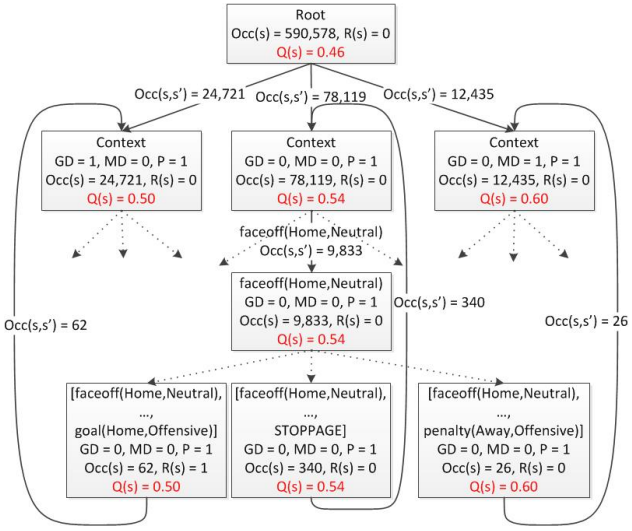


Figure 1: State Transition Graph

Since the complete action history is encoded in the state, action-state pairs are equivalent to state pairs. For example, we can write $Q(s \star a)$ to denote the expected reward from taking action a in state s, where Q maps states to real numbers, rather than mapping action-state pairs to real numbers, as is more usual.

4.4 REWARD FUNCTIONS: NEXT GOAL AND NEXT PENALTY

A strength of Markov Game modelling is value iteration can be applied to many reward functions depending on what results are of interest. We focus on two: scoring the next goal, and receiving the next penalty (a cost rather than a reward). These are two important events that change the course of an ice hockey game. For example, penalties affect goal scoring differentials, as shown in Table 4. Penalties are also one path to goals that a coach may want to understand in more detail. For instance, if a team receives an unusual number of penalties, a coach may want to know which players are responsible and by which actions. The next goal objective can be represented in the Markov Game model as follows.

1. For any state s with a complete play sequence that ends in a Home resp. Away goal, we set $R_H(s) := 1$ resp. $R_A(s) := 1$. For other states the reward is 0.
2. Any state s with a complete play sequence that ends in a Home resp. Away goal is an absorbing state (no transitions from this state).

With these definitions, the expected reward represents the probability that if play starts in state s, a random walk through the state space of unbounded length ends with a goal for the Home team resp. the Away team. The cost function for Receiving the Next Penalty can be represented in exactly the same way.

5 CONSTRUCTING THE STATE TRANSITION GRAPH

The main computational challenge is to build a data structure for managing the state space. The state space is large because each (sub)sequence of actions defines a new state. Since we are modelling the actual hockey dynamics in the "on policy" setting, we need consider only action sequences observed in some NHL match, rather than the much larger space of all possible action sequences. We use the classic AD-tree structure [Moore and Lee, 1998] to compute and store sufficient statistics over observed action sequences. The AD-tree is a tree of play sequences where a node is expanded only with those successors observed in at least one match. The play sequence tree is augmented with additional edges that model further state transitions; for example, a new action sequence is started after a goal. The augmented AD-tree structure compactly manages sufficient statistics, in this case state transition probabilities. It also supports value iteration updates very efficiently.

We outline an algorithm for Context-Aware State Transition Graph construction. The root node initializes the graph, and is an empty node with no context or event information. For each node, the context information GD, MD, and P are set when the new node is created, and the new action a is added to the sequence along with the zone Z that a occurs in. The reward $R(s)$ is also applied to each node. The node counts $Occ(s)$ and edge counts $Occ(s, s')$ are applied to each node and edge respectively, and are used to generate transition probabilities TP for the value iteration using observed frequencies. The NHL play-by-play event data records goals, but no separate event for the shot leading to the goal exists. Following [Schuckers and Curro, 2013], we record the shot leading to the goal in addition to the goal itself by injecting a shot event into the event sequence prior to the goal.

6 VALUE ITERATION

Recall that since states encode action histories, in our model learning the expected value of states is equivalent to learning a Q-function (Section 4.3). In reinforcement learning terms, there is no difference between the value function V and the Q-function in our model. We can therefore apply standard value iteration over states [Sutton and Barto, 1998] to learn a Q-function for our ice hockey Markov Game. Algorithm 1 shows pseudo-code. We compute separate Q-functions for the Home team and for the Away team. Since we are in the "on policy" setting, we have a fixed policy for the other team. This means we can treat the other team as part of the environment, and reduce the Markov Game to two single-agent Markov decision processes. In our experiments, we use a relative convergence of 0.0001 as our convergence criterion, and 100,000 as the maximum number of steps.

Algorithm 1 Dynamic Programming for Value Iteration

Require: Markov Game model, convergence criterion c, maximum number of iterations M

$lastValue = 0$
$currentValue = 0$
$converged = false$
for $i = 1; i \leq M; i \leftarrow i + 1$ **do**
  for all states s in the Markov Game model **do**
    if $converged == false$ **then**
      $Q_{i+1}(s) = R(s) + \frac{1}{Occ(s)} \sum_{(s,s') \in E} (Occ(s, s') \times Q_i(s'))$
      $currentValue = currentValue + |Q_{i+1}(s)|$
    end if
  end for
  if $converged == false$ **then**
    if $\frac{currentValue - lastValue}{currentValue} < c$ **then**
      $converged = true$
    end if
  end if
  $lastValue = currentValue$
  $currentValue = 0$
end for

7 EVALUATION AND RESULTS

We discuss the results of action values in Section 7.1 and player values in Section 7.2. Our state transition graph is evaluated in Section 8.2.

7.1 ACTION IMPACT VALUES

The main quantity we consider is the **impact** of an action as a function of context (= Markov state). This is defined as follows:

$$impact(s, a) \equiv Q_T(s \star a) - Q_T(s)$$

where T is the team executing the action a. In a zero-sum game, the state value is usually defined as the final result following optimal play [Russell and Norvig, 2010]. Intuitively, the value specifies which player has a better position in a state. Since we are not modelling optimal play, but actual play in an "on policy" setting, the expected difference in rewards is the natural counterpart. The impact quantity measures how performing an action in a state affects the expected reward difference. Figure 2 shows a boxplot for the action impact values as they range over different contexts, i.e., states in the Markov Game model. (Boxplots produced with MATLAB R2014a.) The central mark is the median, and the edges of the boxes are the 25th and 75th percentiles. The whiskers are the default value, approximately 2.7 standard deviations. The red dots are outliers beyond 2.7 s.d. and the asterisks are the values given for each action in [Lock and Schuckers, 2009]. A cutoff of -0.2 and 0.2, represented by the horizontal dashed line, was used for the impact values on both boxplots. While the Q-values are based on the frequency of states, we weight all states equally in discussing the properties of the Q-function. The boxplot does not include Q-values for states whose frequency is below 5%. It is clear from Figure 2 that *depending on the context and event history, the value of an action can vary greatly.* The context-dependence is observed for both scoring goals and receiving penalties.

Impact on Scoring the Next Goal. All actions, with the exception of faceoffs won in the offensive zone, have at least one state where the action has a positive impact, and another state with a negative impact. Specific examples of context-dependence that can be found by examining states with extreme impact values include the following.

(1) After blocking the first shot on net when killing a

penalty, the shooting team tends to score more goals ($impact = -0.0864$). But after blocking the second shot on net, the blocking team tends to score more goals ($impact = 0.1399$).

(2) Receiving a penalty when on the powerplay is very bad ($impact = -0.1789$), but if a player on the penalty kill can goad their opponent into an offsetting penalty, it is good ($impact = 0.0474$).

The THoR player ratings compute the impact of actions based on goals that immediately follow the action ([Lock and Schuckers, 2009; Schuckers et al., 2011]; see Section 2). The values given for each action in [Lock and Schuckers, 2009] are displayed as an asterisk in Figure 2(a). The THoR values agree with our median impact values in terms of whether an action generally has positive or negative impact. For example, penalties are known to generally be good for the opposing team, and shots are good for the shooter's team. THoR values are close to the median Markov model values in 6 out of 10 cases. This comparison suggests that THoR aggregates action values over many contexts that the Markov game models explicitly.

Impact on Receiving Penalties. The range of action values with the probability of the next penalty as the objective function is shown in Figure 2(b). Faceoffs in the Offensive Zone and takeaways cause penalties for the opponent. Giveaways and goals tend to be followed by a penalty for the player's team. This finding is consistent with the observation that there are fewer penalties for teams with higher leads [Schuckers and Brozowski, 2012]. A possible explanation is referees are reluctant to penalize a trailing team.

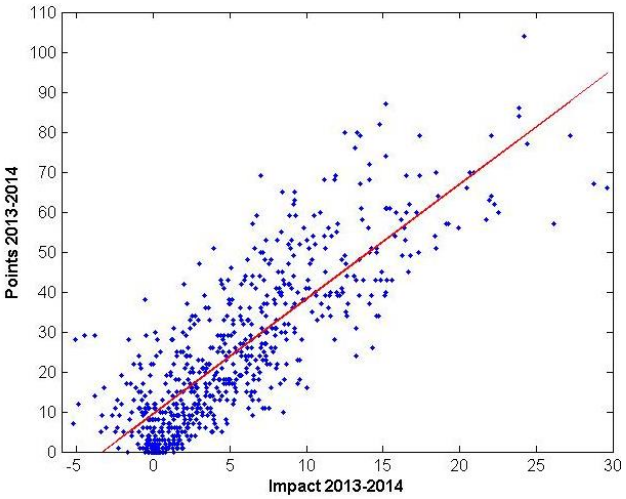


Figure 3: 2013-2014 Player Goal Impact Vs. Season Points

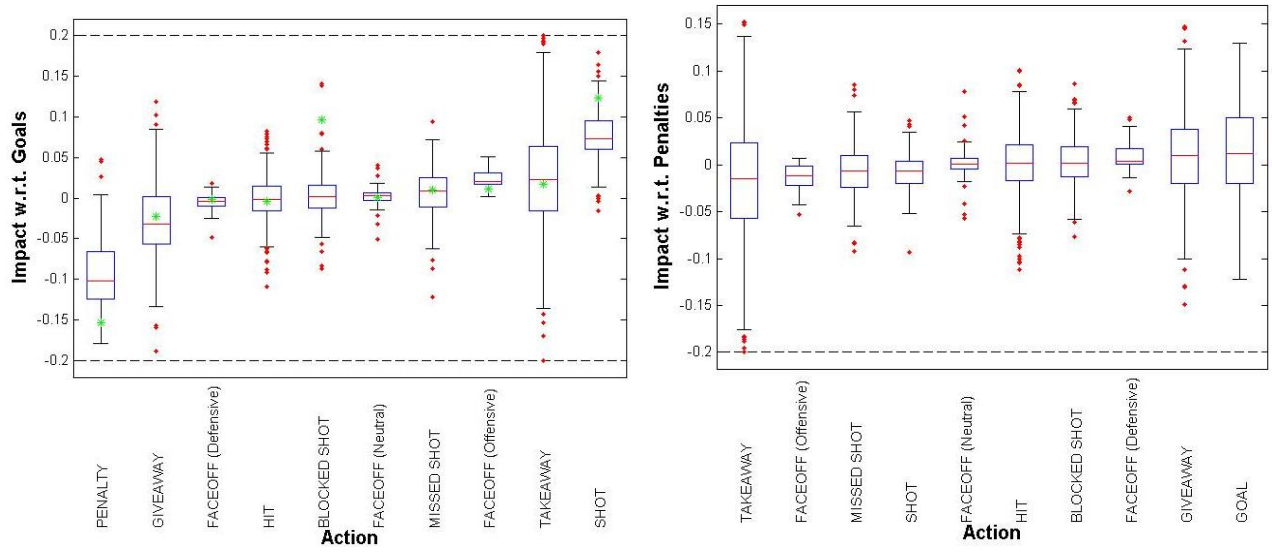
Table 6: 2013-2014 Top-8 Player Impact Scores For Goals

Name	Goal Impact	Points	+/-	Salary
Jason Spezza	29.64	66	-26	$5,000,000
Jonathan Toews	28.75	67	25	$6,500,000
Joe Pavelski	27.20	79	23	$4,000,000
Marian Hossa	26.12	57	26	$7,900,000
Patrick Sharp	24.43	77	12	$6,500,000
Sidney Crosby	24.23	104	18	$12,000,000
Claude Giroux	23.89	86	7	$5,000,000
Tyler Seguin	23.89	84	16	$4,500,000

7.2 PLAYER VALUATIONS

As players perform actions on behalf of their team, it is intuitive to apply the impact scores of team actions to the players performing the action, yielding player valuations. To calculate player valuations, we apply the impact of an action to the player as they perform the action. Next, we sum the impact scores of a player's actions over a single game, and then over a single season, to compute a net season impact score for the player. This procedure is equivalent to comparing the actions taken by a specific player to those of the league-average player, similar to previous work [Pettigrew, 2015; Cervone et al., 2014]. We compare impact on Next Goal Scored with three other player ranking metrics: points earned, salary, and +/-. To avoid confounding effects between different seasons, we use only the most recent full season, 2013-2014. Player impact scores are shown in Table 6. Tables for all seasons are available as well [Routley, 2015]. Figure 3 shows that next goal impact correlates well with points earned. A point is earned for each goal or assist by a player. Since these players have a high impact on goals, they also tend to have a positive +/- rating. Jason Spezza is an anomaly, as he has the highest impact score but a very negative +/- score. This is due to his team performing poorly overall in the 2013-2014 season, and the team overall had a goal differential of -29, one of the highest goal differentials that season. This example shows that impact scores distinguish a player who generally performs useful actions but happens to be on a poor team.

In Table 7, we see player impact with respect to Next Penalty Received. High impact numbers indicate a tendency to cause penalties for a player's own team, or prevent penalties for the opponent. We compare the Q-function impact numbers to Penalties in Minutes (PIM), +/-, and salary. Players with high Q-function numbers have high penalty minutes as we would expect. They also have low +/-, which shows the importance of penalties for scoring chances. Their salaries tend to be lower. There are however notable exceptions, such as Dion Phaneuf, who draws a high salary although his actions have a strong tendency to incur penalties.



(a) Impact on the probability of Scoring the Next Goal. Higher numbers are *better* for the team that performs the action.

(b) Impact on the probability of Receiving the Next Penalty. Higher numbers are *worse* for the team that performs the action.

Figure 2: Action Impact Values vary with context. The central mark is the median, the edges of the box are the 25th and 75th percentiles. The whiskers are at the default value, approximately 2.7 s.d.

Table 7: 2013-2014 Top-8 Player Impacts For Penalties

Name	Penalty Impact	PIM	+/-	Salary
Chris Neil	62.58	211	-10	$2,100,000
Antoine Roussel	54.26	209	-1	$625,000
Dion Phaneuf	52.52	144	2	$5,500,000
Zac Rinaldo	48.65	153	-13	$750,000
Rich Clune	47.08	166	-7	$525,000
Tom Sestito	46.34	213	-14	$650,000
Zack Smith	44.55	111	-9	$1,500,000
David Perron	42.49	90	-16	$3,500,000

8 LESION STUDY

We evaluate the components of our full model by considering simpler models.

8.1 USING AVERAGE ACTION VALUES

We compare context-aware action values vs. fixed action values (as in THoR) in terms of the entropy of the Next Goal conditional probabilities. This quantifies the information lost by ignoring context. Table 8 shows the average entropy for context-aware and context-unaware probabilities. The *context-unaware* Next Goal probability for an action event, is the marginal probability obtained from action-state probabilities by averaging over all states where the action is taken. For all action events, this marginal probability of Next Goal Away is between 47% and 48%. This leads to an average context-unaware entropy of 0.9741 with standard deviation of only 0.0012. The average of the context-aware entropies is 0.9582, which is a lower uncertainty than the context-unaware model; but these entropies show considerable variance, ranging smoothly from 0 to 1, with a large standard deviation of 0.1482. The large standard deviation demonstrates that including context in the model causes the predictive uncertainty of each state to vary more than a context-unaware model, however, it tends to create higher predictive accuracy with a lower average entropy. The entropy results are statistically significant according to the paired t-test ($p = 2.8 \times 10^{-8}$).

Table 8: Context-Aware vs. Context-Unaware Entropies.

Action	Context-Unaware Probability Of Next Goal	Context-Unaware Entropy	Average Context-Aware Entropy
Blocked Shot	0.4840	0.9993	**0.9455**
Faceoff (Defensive)	0.4828	0.9991	**0.9913**
Faceoff (Neutral)	0.5025	1.0000	**0.9944**
Faceoff (Offensive)	0.5335	0.9968	**0.9876**
Giveaway	0.4907	0.9997	**0.9271**
Hit	0.4985	1.0000	**0.9462**
Missed Shot	0.5178	0.9991	**0.9413**
Penalty	0.4442	0.9910	**0.9833**
Shot	0.5673	0.9869	**0.8951**
Takeaway	0.5125	0.9995	**0.9279**
Average Entropy Over Actions		0.9971	**0.9540**

8.2 EXAMINING PROPAGATION EFFECTS

The transition graph construction algorithm facilitates changing the possible state transitions. We utilize this in our experiments to study how different propagation models affect the impact of actions on Next Goal Scored. Specifically, we consider three different transitions graphs of in-

creasing density, their sizes shown in Table 9. The number of states/nodes 1,325,809 is the same for all graphs.

Local Transitions Only State transitions occur only within a play sequence, not across play sequences.

Penalty Transitions State transitions occur from penalty leaf nodes to successor context nodes.

Full Transition Graph Includes loopback edges from all leaf nodes to context nodes, as defined in Section 4.2.

Table 9: Size of State Transition Graphs

	Local	**Penalty**	**Full**
Number of Edges	1,325,808	1,382,780	1,662,504

Action impact changes value depending on the state transition graph. The average differences in action values of the same states across different transition graphs, as well as the standard deviation of the differences, are shown in Table 10. The table shows that the impact on who scores the next goal changes as more information is propagated between states.

Penalty vs. Local. With the local transition graph, value iteration computes the impact of an action on the current play sequence only. Thus the Q-value differential for context states, with the initial empty play sequence, can be obtained from Table 4. The penalty transition graph propagates to the next sequence the effect of penalties only. This means that if a play sequence ends with an event other than a penalty or goal (e.g., stoppage) the penalty transition model treats the play as ending with that event. Thus there is no propagation of the effect of a penalty. In hockey terms, with the local transition graph, the model is not aware that a penalty is followed by a powerplay. Propagating the effect of penalties changes most the estimation of the impact of penalties. This change reflects that receiving a penalty lowers the chances of scoring the next goal. Less obviously, winning a faceoff in the offensive zone has a relatively high positive indirect impact on scoring the next goal, via increasing the probability of a penalty against the opposing team. The effect of winning an offensive zone faceoff can also be seen in Figure 2(b).

Full vs. Penalty. In hockey terms, with the penalty transition graph, the model is aware that a penalty is followed by a *single* powerplay sequence. But if more than one sequence occurs in the same powerplay, the second sequence is ignored in the lookahead. The full transition graph propagates the information about the manpower advantage, until the context features are updated when the dynamics reaches a new context state. Comparing the full transition graph with penalty propagation only, we still find the strongest average impact change for penalties. The simplest explanation of this result is that often in hockey, the effect of penalties goes beyond a single play sequence, and the full transition graph captures more of this medium-term effect.

While the aggregate differential effects show that more propagation leads to more informative results, they do not reflect the considerable context dependence shown by the standard deviations of the impact differentials.

Table 10: Action Impact *Differences* For The Next Goal Depending on Propagation Model.

	Full vs. Penalty		**Penalty vs. Local**	
Action	**Average**	**Std. Dev.**	**Average**	**Std. Dev.**
Blocked Shot	0.0001	0.0210	-0.0003	0.0126
Faceoff (Defensive)	-0.0030	0.0455	-0.0018	0.0225
Faceoff (Neutral)	0.0013	0.0464	0.0006	0.0203
Faceoff (Offensive)	0.0038	0.0432	0.0024	0.0260
Giveaway	-0.0003	0.0245	-0.0001	0.0142
Hit	0.0000	0.0194	-0.0001	0.0126
Missed Shot	-0.0001	0.0218	0.0003	0.0130
Penalty	**-0.0190**	0.0278	**-0.0235**	0.0337
Shot	0.0002	0.0191	0.0002	0.0103
Takeaway	0.0006	0.0245	0.0003	0.0146

9 CONCLUSION

We have constructed a Markov Game Model for a massive set of NHL play-by-play events with a rich state space.Tree-based data structures support efficient parameter estimation and storage. Value iteration computes the values of each action given its context and sequence history—the Q-function of the model. Compared to previous work that assigns a single value to actions, the Q-function incorporates two powerful sources of information for valuing hockey actions: (1) It takes into account the context of the action, represented by the Markov Game state. (2) It models the medium-term impact of an action by propagating its effect to future states. Propagating action effects across sequences utilizes the ordering play sequences in a game, rather than treating sequences as an unordered independent set. Analysis of the computed Q-function shows the impact of an action varies greatly with context, and medium-term ripple effects make a difference. We apply our model to evaluate the performance of players in terms of their actions' total impact. Action impact scores are calculated for players with respect to different objective functions. Impact scores for the next goal correlate with points and +/- statistics. The impact of players on the next penalty has to our knowledge not been previously considered, and shows some surprises, as some highly-paid players hurt their team by causing penalties. In sum, the Q-function is a powerful AI concept that captures much information about hockey dynamics as the game is played in

the NHL.

Future Work The NHL data provides a rich dataset for real-world event modelling. A number of further AI techniques can be applied to utilize even more of the available information than our Markov Game model does. A promising direction is to extend our Markov Game model, which is discrete with data about continuous quantities. These include (i) the time between events, (ii) the absolute game time of the events, (iii) location of shots [Krzywicki, 2005]. Our use of reinforcement learning techniques has been mainly for finding patterns in a rich data set, in the spirit of descriptive statistics and data mining. Another goal is to *predict* a player or team's future performance based on past performance using machine learning techniques. For example, sequence modelling would be able to generalize from play sequence information. A promising model class are Piecewise Constant Conditional Intensity Models for continuous time event sequences [Gunawardana et al., 2011; Parikh et al., 2012]. These models are especially well suited for sequences with a large set of possible events, such as our action events. Another extension is to evaluate players with respect to similar players [Cervone et al., 2014], for instance players who play the same position.

A potential future application for improving play and advising coaches is in finding strengths and weaknesses of teams: We can use the Q-function to find situations in which a team's mix of actions provides a substantially different expected result from that of a generic team. We leave this application for future work.

Acknowledgements

This work was supported by a Discovery Grant from the National Sciences and Engineering Council of Canada. We received helpful comments from Tim Swartz and anonymous UAI referees. Zeyu Zhang assisted with the preparation of plots.

References

Buttrey, S., Washburn, A., and Price, W. (2011). Estimating nhl scoring rates. *Journal of Quantitative Analysis in Sports*, 7(3).

Cervone, D., DAmour, A., Bornn, L., and Goldsberry, K. (2014). Pointwise: Predicting points and valuing decisions in real time with nba optical tracking data. In *8th Annual MIT Sloan Sports Analytics Conference, February*, volume 28.

Gramacy, R., Jensen, S., and Taddy, M. (2013). Estimating player contribution in hockey with regularized logistic regression. *Journal of Quantitative Analysis in Sports*, 9:97–111.

Gunawardana, A., Meek, C., and Xu, P. (2011). A model for temporal dependencies in event streams. In *Advances in Neural Information Processing Systems*, pages 1962–1970.

Hirotsu, N., Wright, M., et al. (2002). Using a markov process model of an association football match to determine the optimal timing of substitution and tactical decisions. *Journal of the Operational Research Society*, 53(1):88–96.

Krzywicki, K. (2005). Shot quality model: A logistic regression approach to assessing nhl shots on goal.

Levesque, H., Pirri, F., and Reiter, R. (1998). Foundations for the situation calculus. *Linköping Electronic Articles in Computer and Information Science*, 3(18).

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the eleventh international conference on machine learning*, volume 157, pages 157–163.

Lock, D. and Schuckers, M. (2009). Beyond +/-: A rating system to compare nhl players. Presentation at joint statistical meetings.

Macdonald, B. (2011). An improved adjusted plus-minus statistic for nhl players.

Moore, A. W. and Lee, M. S. (1998). Cached sufficient statistics for efficient machine learning with large datasets. *J. Artif. Intell. Res. (JAIR)*, 8:67–91.

National Hockey League (2014). National hockey league official rules 2014-2015.

Parikh, A. P., Gunawardana, A., and Meek, C. (2012). Conjoint modeling of temporal dependencies in event streams. In *UAI Bayesian Modelling Applications Workshop*.

Pettigrew, S. (2015). Assessing the offensive productivity of nhl players using in-game win probabilities. In *9th Annual MIT Sloan Sports Analytics Conference*.

Routley, K. (2015). A markov game model for valuing player actions in ice hockey. Master's thesis, Simon Fraser University.

Routley, K., Schulte, O., and Zhao, Z. (2015). Q-learning for the nhl. http://www.cs.sfu.ca/~oschulte/sports/.

Russell, S. and Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

Schuckers, M. and Brozowski, L. (2012). Referee analytics: An analysis of penalty rates by national hockey league officials. In *MIT Sloan Sports Analytics Conference*.

Schuckers, M. and Curro, J. (2013). Total hockey rating (thor): A comprehensive statistical rating of national hockey league forwards and defensemen based upon all on-ice events. In *7th Annual MIT Sloan Sports Analytics Conference*.

Schuckers, M. E., Lock, D. F., Wells, C., Knickerbocker, C. J., and Lock, R. H. (2011). National hockey league skater ratings based upon all on-ice events: An adjusted minus/plus probability (ampp) approach. Unpublished manuscript.

Schumaker, R. P., Solieman, O. K., and Chen, H. (2010). Research in sports statistics. In *Sports Data Mining*, volume 26 of *Integrated Series in Information Systems*, pages 29–44. Springer US.

Sidhu, G. and Caffo, B. (2014). Moneybarl: Exploiting pitcher decision-making using reinforcement learning. *The Annals of Applied Statistics*, 8(2):926–955.

Spagnola, N. (2013). The complete plus-minus: A case study of the columbus blue jackets. Master's thesis, University of South Carolina.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement learning : an introduction*. MIT Press, Cambridge, Mass.

Thomas, A., Ventura, S., Jensen, S., and Ma, S. (2013). Competing process hazard function models for player ratings in ice hockey. *The Annals of Applied Statistics*, 7(3):1497–1524.