18

A Unified Theory of Locomotion

Graham Pentheny

- 18.1 Introduction
- 18.2 Unified Theory of Locomotion
- 18.3 Positional Restrictors and Directors
- 18.4 Velocity Restrictors and Directors
- 18.5 Cascading Value Dependencies

- 18.6 Combining Movement Systems
- 18.7 Value Arbitration and Combination
- 18.8 Conclusion
- References

18.1 Introduction

AI character movement in games is a complex problem with many disparate solutions. Character movement techniques must account for many considerations and constraints. Different movement solutions for games address different aspects of the problem, necessitating a combination of complementary solutions for robust character movement. Often, combining multiple movement systems results in complex, intricate, and brittle logic that is both difficult to debug and reuse.

The unified theory of locomotion is a common language by which all movement systems can be defined. Common systems such as velocity obstacle collision avoidance, steering behaviors, and PID controllers can be viewed as specific applications of this theory, with their own individual strengths. Viewing movement systems in this way can help solve many issues faced in aggregating multiple movement systems together. The unified theory of locomotion is a common language for movement that can provide a basis for improving AI movement systems to be more robust to design changes, debugging, and experimentation.

18.2 Unified Theory of Locomotion

Character movement in games can be viewed as a series of steps, each answering a different question about the desired behavior. First and most fundamentally, the character must have a goal location. This is often determined by domain-specific high-level reasoning. Once a destination is established, the character must figure out the best way to move through the world to the destination. Path-planning algorithms are designed to solve this problem, breaking a larger positional goal into a series of shorter subgoals. Next, the character must determine what dynamic obstacles, if any, exist that would prevent them from getting to their next subgoal. These obstacles must be avoided, while maintaining a course toward the character's current subgoal. Finally, the character must move itself through the world, holding as close as possible to an ideal velocity.

The unified theory of locomotion breaks down movement solutions into the component values of motion that they influence and the type of influence they impart on each value. Movement systems are categorized as affecting one or more of the character's position, velocity, or acceleration values. Additionally, each system is categorized as either restricting or directing each value they influence. In this way, we can describe systems with terms like "velocity restrictor" and "acceleration director."

Systems that are defined as a value restrictor remove allowed values from the possible range of outputs. For example, a velocity restrictor generates a set of velocity vectors that the agent is not allowed to follow. A value director, conversely, determines a specific desired value for a movement attribute. Steering behaviors, for example, are generally "velocity directors," each outputting a single, ideal velocity value. By categorizing behaviors by the attributes they affect, and whether they direct or restrict them, we can easily compare two or more movement systems objectively.

18.3 Positional Restrictors and Directors

The simplest influencer on a character's movement is the environment that they move within. A simplified representation of the walkable area is usually generated in the form of a navmesh (Tozour 2004). A navmesh dictates which areas the character is able to move in and thus restricts the positions that a character can have in an environment. Because it is narrowing the infinite domain of positional values to a small subset, it is considered a "positional restrictor."

Conversely, path planning takes a goal location and directs the movement of a character though the environment toward that goal. Path planning determines a sequence of positional subgoals, defining the ideal movement direction and position of the character. Because it provides a specific position, or sequence of positions, it is considered a "positional director."

18.4 Velocity Restrictors and Directors

Reciprocal velocity obstacles (RVO) (van den Berg et al. 2011) are a common approach to local collision avoidance. This technique avoids localized collisions with obstacles and other characters by disallowing velocities that would result in a collision with another character or obstacle. As velocity obstacle approaches directly limit the allowed velocity values, velocity obstacles are an ideal example of a "velocity restrictor."

PID controllers (Melder and Tomlinson 2014) are a signal feedback control mechanism that is often used to control character velocity in games. PID controllers modify a character's velocity incrementally, attempting to smoothly match it to a goal velocity. Often, a character's ideal velocity will be computed as their fastest move speed along the direction of the path to their goal. PID controllers augment the character's velocity incrementally to match this ideal computed velocity. As PID controllers output a single velocity that respects the current set of velocity restrictions, PID controllers are considered a type of "velocity director."

18.5 Cascading Value Dependencies

An important aspect of the unified theory of locomotion is the relationship between restrictors and directors across physical values. Effects on lower level movement attributes indirectly affect higher level derived values. For example, positional restrictors also indirectly affect the velocities that the character is allowed to follow. Velocities that move the character into a disallowed position should also be disallowed. In a similar way, velocity restrictors also indirectly affect the domain of allowed accelerations.

The opposite, however, does not hold true. Velocity value restrictions should not affect the positions a character can reach. For example, certain types of characters may not be able to move backward very quickly. Although this characteristic imparts a restriction on the set of allowed velocities, it does not affect the positions where the character is allowed to be. Additionally, a character might have certain restrictions on the magnitude that it is capable of accelerating in a given direction. This should not affect what velocities the character is able to reach, just the steps necessary to achieve their desired velocity. This assertion, however, does not hold if the character is only able to accelerate in a single direction. This is a rare case that would necessitate special consideration and will not be addressed.

18.6 Combining Movement Systems

Often, combining multiple movement systems is difficult because each system is designed and tested in isolation. Combining multiple systems is greatly simplified by applying the unified theory of locomotion as a context for viewing movement systems. In this context, combining multiple systems simply consists of separately combining the restrictions and directions that each system imparts on each movement value. For example, context steering (Fray 2015) generates a map of velocity values and weights the benefit of each according to a set of rules. RVO performs a similar action, restricting sets of velocities that would result in a collision with an obstacle. As RVO and context steering are both velocity restrictors, we can compute the union of their velocity domain restrictions to combine their effects. This combination reaps the benefits of both systems; however, it simplifies them to a single result.

18.7 Value Arbitration and Combination

Combining multiple movement systems necessitates combining the restrictions and directions that each system imparts on each physical value. Combining restrictors simply involves generating the union of the sets of restrictions they impart on a physical value.

The process of combining value directors, however, is more involved. It consists of combining individually computed physical values, while respecting the limitations imparted by the restrictors. The director combination process can be broken down into two steps: high-level arbitration between directors and low-level combination of the resulting values.

High-level director arbitration determines which director techniques are combined to drive a character. High-level arbitration observes the character, its position, and its relationship to the other characters and the environment and makes a judgment about which value directors should affect the character and with what precedence. For example, a character with many neighbors might prefer a collision avoidance director over a formation-following director, as avoiding collisions is considered more important than maintaining formation. High-level arbitration is a subtle, domain-specific process that has a significant impact on the resulting character behavior. Designing an arbitration system involves understanding the behavioral considerations of a character, and how different systems address those considerations.

Low-level combinators are systems that combine the results of multiple value directors. Different techniques exist for combining value directors, and like high-level arbitrators, designing effective combinators involves understanding the problem domain and the character's desired behavior. A common low-level combinator approach is to prioritize each value director and combine their resulting values in priority order until a threshold of total influence is met. This ensures that important behaviors are more impactful on character behavior than others. This straightforward approach is often utilized successfully in the context of steering behaviors (Reynolds 1987).

In addition to prioritized summation, value directors can additionally be weighted, artificially amplifying their output. The result of each director is scaled by a specific, corresponding value before being combined. This approach gives control over how sensitive a specific system should be on a character and can be used to create unique movement styles or personalities.

18.8 Conclusion

The unified theory of locomotion provides a context for viewing and reasoning about character movement. It provides a common language that can describe the behavior of different movement systems. Each system can be broken down into either directing or restricting one of the physical movement values of a character. These directors and restrictors on low-level movement values also indirectly affect derived values.

Applying the unified theory of locomotion helps simplify and direct the combination of multiple movement systems. Value restrictors are combined by taking the logical union of their results, while combining directors are broken into high-level arbitration and low-level combination. High-level arbitration determines which value directors affect a character and in what precedence. Low-level combination aggregates the resulting values in a way that prioritizes more important behaviors while giving AI designers the ability to control the characteristics of a character's movement behavior.

References

- Fray, A. 2015. Context steering: Behavior-driven steering at the macro scale. In *Game AI Pro 2*, ed. S. Rabin. Boca Raton, FL: CRC Media, pp. 173–181.
- Melder, N., and S. Tomlinson. 2014. Racing vehicle control systems using PID controllers. In *Game AI Pro*, ed. S. Rabin. Boca Raton, FL: CRC Media, pp. 491–500.
- Reynolds, C. 1987. Flocks, herds and schools: A distributed behavioral model. *International Conference and Exhibition on Computer Graphics and Interactive Techniques*, Anaheim, CA: Association for Computing Machinery, pp. 25–34.
- Tozour, P. 2004. Search space representations. In *AI Game Programming Wisdom 2*, ed. S. Rabin. Hingham, MA: Charles River Media, pp. 85–102.
- van den Berg, J., S. J. Guy, M. Lin, and D. Manocha. 2011. Reciprocal n-body collision avoidance. *Springer Tracts in Advanced Robotics*, 70: 3–19.