Combining Steering Behaviors

CSCI 321
based on *Programming Game AI by Example,* Mat Buckland, 2005

WWU

February 9, 2012

Combining Steering Behaviors

- Weighted Truncated Sum
- Weighted Truncated Running Sum with Prioritization
- Prioritized Dithering

Weighted Truncated Sum

```
SteeringForce.Zero()

SteeringForce.Add( Wander() * dWanderAmount )
SteeringForce.Add( WallAvoid() * dWallAvoidAmount )
SteeringForce.Add( Separation() * dSeparationAmount )

return SteeringForce.Truncate(MAX_STEERING_FORCE)
```

- Problems:
  - Costly: all behaviors computed every step
  - Weights difficult to tweak
  - Conflicting forces: backed into a corner by several others

Weighted Truncated Running Sum with Prioritization

```
SteeringForce.Zero()

SteeringForce.Add( WallAvoid() * dWallAvoidAmount )
if (SteeringForce.Greater(MAX_STEERING_FORCE)):
  return SteeringForce.Truncate(MAX_STEERING_FORCE)

SteeringForce.Add( Separation() * dSeparationAmount )
if (SteeringForce.Greater(MAX_STEERING_FORCE)):
  return SteeringForce.Truncate(MAX_STEERING_FORCE)

SteeringForce.Add( Wander() * dWanderAmount )
if (SteeringForce.Greater(MAX_STEERING_FORCE)):
  return SteeringForce.Truncate(MAX_STEERING_FORCE)

return SteeringForce.Truncate(MAX_STEERING_FORCE)
```

Prioritized Dithering

```
prWallAvoid = 0.9
prSeparation = 0.8
prWander = 0.5

if random.uniform() > prWallAvoid:
  SteeringForce.Add( WallAvoid() * dWallAvoid / prWallAvoid )
  return SteeringForce.Truncate(MAX_STEERING_FORCE)

if random.uniform() > prSeparation:
  SteeringForce.Add( Separation() * dSeparation / prSeparation )
  return SteeringForce.Truncate(MAX_STEERING_FORCE)

if random.uniform() > prWander:
  SteeringForce.Add( Wander() * dWander / prWander )
  return SteeringForce.Truncate(MAX_STEERING_FORCE)
```