Sensors and Measurement Techniques for Position Recovery and Object Localization

Emil M. Petriu, Dr. Eng., FIEEE
Professor
School of Information Technology and Engineering
University of Ottawa
Ottawa, ON., K1N 6N5 Canada
petriu@site.uottawa.ca
Position Sensors - classified according to their range

1. **contact sensors**
   * make/break contact
   * tactile probes
   * analog position sensors
   * position encoders

2. **“near to”, or proximity, sensors**
   * close range sensors:
     - time-of flight (sonar, IR, radar)
     - triangulation
   * imaging
     - laser scanners
     - vision

3. **“far away” sensors**
   * long range sensors:
     - time-of flight (sonar, IR, radar)
     - triangulation
   * imaging
     - vision
     - radar
     - IR
The most common position transducers are: potentiometers, synchros and resolvers, encoders, RVDT (rotary variable differential transformer) and INDUCTOSYN.

Encoders are digital position transducers which are the most convenient for computer interfacing.

Incremental encoders are relative-position transducers which generate a number of pulses proportional with the traveled rotation angle. They are less expensive and offer a higher resolution than the absolute encoders. As a disadvantage, incremental encoders have to be initialized by moving them in a reference ("zero") position when power is restored after an outage.

Absolute encoders are attractive for joint control applications because their position is recovered immediately and they do not accumulate errors as incremental encoders may do. Absolute encoders have a distinct n-bit code (natural binary, Gray, BCD) marked on each quantization interval of a rotating scale. The absolute position is recovered by reading the specific code written on the quantization interval currently facing the reference marker.
The metrological performance of a position recovery system three parameters: **accuracy**, **repeatability**, and **resolution**.

**Accuracy** is the difference between the actual location and the recovered location.

**Repeatability** is the variation in the recovered location.

**Resolution** is the minimum distance that the measurement system can detect.

The repeatability and resolution of most system is better than their accuracy.
INCREMENTAL OPTICAL ENCODERS

Light Source

Photo Detector

Mask

On

Off

Scale with Markings

Light Source

Photo Detector

Mask

On

Off

Scale with Markings
The quantization intervals on the encoded track are conveniently marked in such a way that the code read for each quantization interval is unique being able to unequivocally identify each interval. An a priori defined “Code/Position” mapping (mathematical function, or look-up table) is then used to recover the actual position of the quantization interval currently probed. The most popular codes and associate probe (reading head) technology are binary. However there are now new computer vision, radio beacon, etc. probing technologies that allow multi-valued (polyvalent) marking of the encoded track.
The straightforward approach for the absolute position encoding requires that each quantization interval of a scale be marked with a distinct n-bit code.
**Absolute Natural Binary Encoder.**

The position $P$ of the pointer with respect to the origin of this scale is estimated by reading the specific code \{x(k)? \{0,1\} / k=1,...,4\} written on the quantization interval currently facing the pointer:

$$P = p \cdot q = \left( \sum_{k=1}^{4} x(k) \cdot 2^{k-1} \right) \cdot q \quad / \quad x(k) \in \{0,1\}$$

Absolute position encoding using a natural 4-bit code
The (n+1) bit natural binary absolute encoder. A total of n+1 reading heads are used, but only a number of n code tracks (the most significant) have to be physically implemented on the moving scale.
Code reading synchronization logic
for the natural binary absolute encoder
Time diagrams for the code reading synchronization logic
Why natural binary coding cannot be used in practice for absolute position recovery?
A n-bit code would be needed for each quantization step, resulting in n binary tracks in parallel with the guide-path. For instance, the encoding of a 160 m long guide-path with a 0.01 m resolution would need 14 tracks running in parallel with the guide path.
Pseudo-Random Encoding

A practical solution allowing absolute position recovery with any desired n-bit resolution while employing only one binary track, regardless of the value of n.

Table 1  Feedback equations for PRBS generation

<table>
<thead>
<tr>
<th>Shift register length n</th>
<th>Feedback for direct PRBS</th>
<th>Feedback for reverse PRBS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$R(0) = R(n) \oplus c(n-1) \cdot R(n-1) \oplus \ldots \oplus c(1) \cdot R(1)$</td>
<td>$R(n+1) = R(1) \oplus b(2) \cdot R(2) \oplus \ldots \oplus b(n) \cdot R(n)$</td>
</tr>
<tr>
<td>4</td>
<td>$R(0) = R(4) \oplus R(1)$</td>
<td>$R(5) = R(1) \oplus R(2)$</td>
</tr>
<tr>
<td>5</td>
<td>$R(0) = R(5) \oplus R(2)$</td>
<td>$R(6) = R(1) \oplus R(3)$</td>
</tr>
<tr>
<td>6</td>
<td>$R(0) = R(6) \oplus R(1)$</td>
<td>$R(7) = R(1) \oplus R(2)$</td>
</tr>
<tr>
<td>7</td>
<td>$R(0) = R(7) \oplus R(3)$</td>
<td>$R(8) = R(1) \oplus R(4)$</td>
</tr>
<tr>
<td>8</td>
<td>$R(0) = R(8) \oplus R(4) \oplus R(3) \oplus R(2)$</td>
<td>$R(9) = R(1) \oplus R(3) \oplus R(4) \oplus R(5)$</td>
</tr>
<tr>
<td>9</td>
<td>$R(0) = R(9) \oplus R(4)$</td>
<td>$R(10) = R(1) \oplus R(5)$</td>
</tr>
<tr>
<td>10</td>
<td>$R(0) = R(10) \oplus R(3)$</td>
<td>$R(11) = R(1) \oplus R(4)$</td>
</tr>
</tbody>
</table>
A \((2^n-1)\) term Pseudo-Random Binary Sequences (PRBS) generated by a \(n\)-bit modulo-2 feedback shift register is used as an one-bit / quantization-step absolute code. The absolute position identification is based on the PRBS window property. According to this any \(n\)-tuple seen through a \(n\)-bit window sliding over PRBS is unique and henceforth it fully identifies each position of the window.

The figure shows, as an example, a 31-bit term PRBS: 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, generated by a 5-bit shift register. The 5-bit n-tuples seen through a window sliding over this PRBSs are unique and represent a 1-bit wide absolute position code.
Pseudo-Random Binary Sequence (PRBS) encoded track with one bit per quantization step allows recovery of the absolute position of an optically guided Automated Guided Vehicle (AGV)
Serial-parallel code conversion of the absolute position \( p = 18 \) on a 31-position PRBS encoded track with four milestones.

\[ PRBS = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \]
{x(k)=S(p+n-k)|k=n,...,1}  
Pseudo-random n-tuple  
corresponding to the position  
index p = m \cdot t + r  

Serial-parallel pseudo-random / natural code conversion algorithm
Serial-parallel code conversion costs as a function of the distance $t$ between milestones. $k_1$ is the equipment cost associated with each milestone, $k_2$ is the basal hardware cost for the serial back-shift operations, $k_3$ is the basal temporal cost for a fully parallel solution, and $k_4$ is the temporal cost associated with each back-shift operation.
AGV’ S POSITION AS STORED IN THE READING BUFFER

The “naïve” straightforward synchronization of the code readings introduces an “hysteresis error”.

"naïve"
A more effective synchronization method which eliminates the "hysteresis error" and doubles the overall measuring resolution.
Synchronization logic
Implementation details of the PRBS encoded absolute position measurement
Protection circuit against measuring errors that occur when the AGV changes its moving direction on the guide-path.
Optically guided AGV tracking a PRBS encoded guide path
PRBS encoded guide path allows recovery of the absolute position of an AGV using computer vision
Computer vision recognition of the pseudo-random binary code
Wall-mounted PRBS encoded guide path allows recovery of the absolute position of the AGV using computer vision
Folding a Pseudo-Random Binary Sequence (PRBS) to produce a Pseudo-Random Binary Array (PRBA)
Illustrating the window property in a Pseudo-Random Binary Array (PRBA). The 3-by-2 code seen through a window on a 7-by-9 PRBA is unique and used as absolute code for the window position \((i,j)\).
Pseudo-Random Binary Array (PRBA) encoding for the recovery of the 2D absolute position of a free ranging mobile robot using computer vision
AGV
FLOOR
REFERENCE SYSTEM

VIDEO
CAMERA

FLOOR
PORTION
SEEN BY
CAMERA

O_u

O_g

O_r
Recovering the Position and Orientation of 3D Objects
PRBS encoding for computer vision recovery of the 3D position of a probe mapping the electromagnetic-field radiated by a telephone set
Computer vision recovery of the pseudo-random code
Model-based recognition of a pseudo-random encoded object
The geometric model definition of four types of 3-D objects: (a) rectangular parallelepiped, (b) triangular prism, (c) square pyramid, and (d) right circular cylinder.
3D object models are unfolded and mapped on the encoding pseudo-random array.
Pseudo-Random Multi-Valued Sequences (PRMVS)

A more compact absolute position encoding can be obtained by using Pseudo-Random Multi-Valued Sequences (PRMVS) where sequence elements are entries taken from an alphabet with more than two symbols.

Compared to the traditional approach the resulting number of code tracks on the scale at the same resolution decreases proportionally with the size of the alphabet used.

As an example, a two stage shift register, \( n=2 \), having the feedback defined by the primitive polynomial \( h(x) = x^2 + x + A \) over \( GF(4) = \{0, 1, A, A^2\} \), with \( A^2 + A + 1 = 0 \) and \( A^3 = 1 \), generates the 15-term PRMVS \( \{0, 1, A^2, 1, 0, A, A, 1, A, 0, A^2, A^2, A, A^2\} \). Any 2-tuple seen through a 2-position window sliding over this sequence is unique.
A "pseudo-random multi-valued sequence" (PRMVS) has multi-valued entries taken from an alphabet of q symbols, where q is a prime or a power of a prime. Such a \((q^n - 1)\)-term sequence is generated by an \(n\)-position shift register with a feedback path specified by a primitive polynomial

\[
h(x) = x^n + h_{n-1}x^{n-1} + \ldots + h_1x + h_0
\]

of degree \(n\) with coefficients from the Galois field \(GF(q)\).

When \(q\) is prime, the integers modulo \(-q\) form the Galois field \(GF(q) = \{0, 1, 2, \ldots, p-1\}\) in which the addition, subtraction, multiplication and division are carried out modulo \(q\).

When \(q\) is a power of a prime, \(q = pm\), the integers modulo \(q\) do not form a field and the Galois field elements are expressed as the first \(q - 1\) powers of some primitive element, labeled here for convenience by the letter \(A\): \(GF(q) = \{0, 1, A, A^2, \ldots, A^{q-2}\}\).

The primitive polynomials used for different PRMVS generation depend on the nature of the addition/subtraction and multiplication/division tables adopted for each particular Galois field.

A number of primitive polynomials over \(GF(q)\) are given in the next Table for \(GF(3), GF(4), GF(8),\) and \(GF(9)\). It is obvious that the PRBS is a particular case of PRMVS for \(GF(2) = \{0, 1\}\).
<table>
<thead>
<tr>
<th>n</th>
<th>q=3</th>
<th>q=4</th>
<th>q=8</th>
<th>q=9</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$x^2+x+2$</td>
<td>$x^2+x+A$</td>
<td>$x^2+Ax+A$</td>
<td>$x^2+x+A$</td>
</tr>
<tr>
<td>3</td>
<td>$x^3+2x+1$</td>
<td>$x^3+x^2+x+A$</td>
<td>$x^3+x+A$</td>
<td>$x^3+x+A$</td>
</tr>
<tr>
<td>4</td>
<td>$x^4+x+2$</td>
<td>$x^4+x^2+Ax+A^2$</td>
<td>$x^4+x+A^3$</td>
<td>$x^4+x+A^5$</td>
</tr>
<tr>
<td>5</td>
<td>$x^5+2x+1$</td>
<td>$x^5+x+A$</td>
<td>$x^5+x^2+x+A^3$</td>
<td>$x^5+x^2+A$</td>
</tr>
<tr>
<td>6</td>
<td>$x^6+x+2$</td>
<td>$x^6+x^2+x+A$</td>
<td>$x^6+x+A$</td>
<td>$x^6+x^2+Ax+A$</td>
</tr>
<tr>
<td>7</td>
<td>$x^7+x^6+x^4+1$</td>
<td>$x^7+x^2+Ax+A^2$</td>
<td>$x^7+x^2+Ax+A^3$</td>
<td>$x^7+x+A$</td>
</tr>
<tr>
<td>8</td>
<td>$x^8+x^3+2$</td>
<td>$x^8+x^3+x+A$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>$x^9+x^7+x^5+1$</td>
<td>$x^9+x^2+x+A$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>$x^{10}+x^9+x^7+2$</td>
<td>$x^{10}+x^3+A(x^2+x+1)$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following relations apply:

- For $GF(4)=GF(2^2)$: $A^2+1=0$, $A^2=A+1$, and $A^3=1$
- For $GF(8)=GF(2^3)$: $A^3+A+1=0$, $A^3=A+1$, $A^4=A^2+A$, $A^5=A^2+A+1$, $A^6=A^2+1$, and $A^7=1$
- For $GF(9)=GF(3^2)$: $A^2+2A+2=0$, $A^2=A+1$, $A^3=2A+1$, $A^4=2$, $A^5=2A$, $A^6=2A+2$, $A^7=A+2$, and $A^8=1$

According to the PRMVS window property, any q-valued contents observed through a n-position window sliding over the PRMVS is unique and fully identifies the current position of the window.

As an example, a two-stage shift register, $n=2$, having the feedback path defined by the primitive polynomial $h(x)=x^2+x+A$ over $GF(4)=\{0,1,A,A^2\}$, with $A^2+A+1=0$ and $A^3=1$, generates the 15-term PRMVS: $\{0, 1, 1, A^2, 1, 0, A, A, 1, A, 0, A^2, A^2, A, A^2\}$.
Any 2-tuple seen through a 2-position window sliding over this sequence is unique.
Implementation of a PRMVS encoded track using
polyvalent code_markings & reading_heads (probes)

Implementation of a PRMVS encoded track using
binary code_markings & reading_heads (probes)
The **rows** are encoded with the terms of a PRMVS \( \{X(i) | i = 0,1,..., q^n_x -1\} \) generated by a **nx**-stage shift register having entries taken from an alphabet of \( q_x \) symbols.

The **columns** are encoded with the terms of a PRMVS \( \{Y(j) | j = 0,1,..., q^n_y -1\} \) generated by a **ny**-stage shift register and having entries taken from an alphabet of \( q_y \) symbols.

**Absolute position recovery of any grid-node of coordinated \((i,j)\)** needs to identify a **nx-by-ny window**. The row-index \( i \) can be recovered if it is possible to identify a nx-tuple containing \( X(i) \). The column-index \( j \) can be recovered if it is possible to identify a ny-tuple containing \( Y(j) \).
PRMVS grid having 15 row-lines and 15 column-lines encoded with the terms of two PRMVS \( X(i)=Y(i) \) \( i=0,1,...,q^n-1 \) where \( q=4 \) and \( n=2 \), defined over \( GF(4) = \{0,1,A,A^2\} \). Absolute position recovery needs to identify a 2-by-2 window in this case. The row-index \( i \) of a given grid node \( (i,j) \) can be recovered if it is possible to identify the \( X(i) \) and \( X(i+1) \) \([\text{or } X(i-1)]\) associated with two adjacent row lines. The column-index \( j \) of a given grid node \( (i,j) \) can be recovered if it is possible to identify the \( Y(j) \) and \( Y(j+1) \) \([\text{or } Y(j-1)]\) associated with two adjacent column lines.
Recovery of the 3D shape of objects using structured light
Point identification in pseudo-random encoded structured light
Pseudo-Random Binary Array encoded structured-light grid projected on a 3D object
Pseudo-Random Multi Valued Sequence (PRMVS) opportunistically color encoded structured light grid projected on a cube

Recovered corner points at the intersection of grid line edges
Pseudo-Random Multi Valued Sequence (PRMVS)
structured-light grid projected on a 3D object
Multisensor Data Fusion

MULTISENSOR FUSION

FUSION
- Symbol Level
- Feature Level
- Pixel Level
- Signal Level

SEPARATE OPERATION

GUIDING OR CUEING

SENSOR REGISTRATION

SENSOR SELECTION

SENSOR CONTROLLER

SYSTEM CONTROLLER

WORLD MODEL

SENSOR 1
SENSOR 2
SENSOR n

SENSOR MODEL
SENSOR MODEL
SENSOR MODEL
Multisensor integration refers to the “synergistic use of the information provided by multiple sensors to assist the accomplishment of a task.”

Multisensor fusion refers to “any stage in the integration process where there is an actual combination (or fusion) of different sensor information into a unique representational format”.

Advantages of Multiple Sensors

- **Redundancy** - Redundant information is provided from a group of sensors or by a single sensor over time when each sensor observes (possibly with different fidelity), the same features of interest.

- **Complementarity** - Complementary information from multiple sensors allows for the perception of features that are impossible to be observed using just the information from individual sensors operating separately.

- **Timeliness** - More timely information may be provided by multiple sensors due to the actual speed of operation of each sensors, or to the processing parallelism that is possible to be achieved as part of the integration process.

- **Cost** - Integrating many sensors into one system can often use many inexpensive devices to provide data that is of the same, or even superior quality to data from a much more expensive and less robust device.
Mobile robot navigation using multiple IR sensors and vision
IR sensor based triangulation for Pentax Zoom 60x camera
Axial characteristics for the IR sensor
Probabilistic models of the IR sensor for two different measurement ranges
Occupancy grid map of a round wall around the rotating IR sensor after one turn
Occupancy grid map of a round wall around the rotating IR sensor after ten turns
Multi IR sensor system on board the mobile robot
Layout of the room explored by the mobile robot with eight on board IR sensors
The recovered shape of explored room by fusing the data from the eight IR sensors using the probability occupancy grid method
Errors in Multisensor Systems

- **Errors in the Integration and Fusion Process** - a major source of errors when fusing redundant information from multiple sensors is the sensor registration.

- **Errors in the Sensory Information** - usually are assumed to be caused by a random noise (uncorrelated in space or time, Gaussian and independent) that can be adequately modelled as a probability of distribution. The consistency of sensor measurements is increased by eliminating the spurious measurements so that they are not included in the fusion process.

- **Errors in the System Operation** - A multisensor system must have the ability to recognize and recover from sensor failure. Sometimes in unknown environments, it may be difficult or impossible to calibrate sensors. A solution would be the creation of a knowledge database for each sensor permitting an auto-calibration process of the system.
Error characteristics of the IR sensor for two colors of the targets